NetBlazer®

Command Reference Manual

TELEBIT®
First In Dial-Up Networking

NetBlazer™

Command Reference

90273-01 3/23/93

Trademarks

Telebit, NetBlazer, and TrailBlazer are trademarks of Telebit Corporation.

Telebit gratefully acknowledges the following additional copyrights:

- Apple and AppleTalk are registered trademarks of Apple Computer, Inc.
- Novell and NetWare are registered trademarks of Novell, Inc.
- UNIX is a registered trademark of AT&T Bell Laboratories.
- TCL is copyrighted by the University of California, Berkeley (unrestricted redistribution).

Other brands or product names are trademarks or registered trademarks of their respective holders.

Software Copyright Notice

This product contains software which has been copyrighted by Telebit Corporation, Sunnyvale, CA. All rights are reserved.

PLEASE READ THIS DOCUMENT CAREFULLY BEFORE USING THE NETBLAZER. BY USING THE SOFTWARE, YOU ARE AGREEING TO THE TERMS OF THIS LICENSE. IF YOU DO NOT AGREE TO THE TERMS OF THIS LICENSE, PROMPTLY RETURN THE UNUSED SOFTWARE TO THE PLACE OF PURCHASE AND YOUR MONEY WILL BE REFUNDED.

1. License. The NetBlazer software (the "Software") and related documentation are licensed to you by Telebit Corporation. You own the disk on which the Software is recorded but Telebit retains title to the Software. This License allows you to use the Software on a single computer and to make one copy of the Software in machine-readable form for backup purposes and a reasonable number of copies of the following files for your remote access to your NetBlazer: PPPCFG.EXE, PPPCFG.MDM, PPPCFG.HLP, IPX.COM, COMPPP.EXE, IPXPPP.LAN, and WATCHPPP.EXE. You must reproduce on all such copies the copyright notice and any other proprietary legends that were on the original copy of the Software. You may transfer the Software, the backup copy and copy(s) for remote access of the Software, the related documentation and a copy of this License to another party provided the other party reads and agrees to accept the terms and conditions of this License.

2. Restrictions. The Software contains copyrighted material, trade secrets, and other proprietary information and in order to protect them you may not decompile, reverse engineer, disassemble or otherwise reduce the Software to a human-perceivable form. You may not modify, network, rent, lease, loan, sell, distribute or create

derivative works based upon the Software in whole or in part. Use, duplication, or disclosure of the Software by the U. S. Government is subject to restrictions as set forth in subdivision (b) (3) (ii) of the Rights in Technical Data and Computer Software clause at 252.227-7013. Use, duplication, and disclosure by the governments of any other countries is subject to restrictions of similar applicable laws.

3. Termination. This License is effective until terminated. You may terminate this License at any time by destroying the Software and all copies thereof. This License will terminate immediately without notice from Telebit if you fail to comply with any provision of this License.

4. Limited Warranty on Media. Telebit warrants the disks on which the Software is recorded to be free from defects in materials and workmanship under normal use for a period of ninety (90) days from the date of purchase as evidenced by a copy of the receipt. Telebit's entire liability and your exclusive remedy will be replacement of the disk not meeting Telebit's limited warranty and which is returned to Telebit or a Telebit authorized representative with a copy of the receipt. Telebit will have no responsibility to replace a disk damaged by accident, abuse or misapplication.

5. Disclaimer or Warranty on Software. You expressly acknowledge and agree that use of the Software is at your sole risk. The Software and related documentation are provided "AS IS" and without warranty of any kind and Telebit EXPRESSLY DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. TELEBIT DOES NOT WARRANT THAT THE FUNCTIONS CONTAINED IN THE SOFTWARE WILL MEET YOUR REQUIREMENTS, OR THAT THE OPERATION OF THE SOFTWARE WILL BE UNINTERRUPTED OR ERROR-FREE, OR THAT DEFECTS IN THE SOFTWARE WILL BE CORRECTED. FURTHERMORE, TELEBIT DOES NOT WARRANT OR MAKE ANY REPRESENTATIONS REGARDING THE USE OR THE RESULTS OF THE USE OF THE SOFTWARE OR RELATED DOCUMENTATION IN TERMS OF THEIR CORRECTNESS, ACCURACY, RELIABILITY, OR OTHERWISE. NO ORAL OR WRITTEN INFORMATION OR ADVICE GIVEN BY TELEBIT OR A TELEBIT AUTHORIZED REPRESENTATIVE SHALL CREATE A WARRANTY OR IN ANYWAY INCREASE THE SCOPE OF THIS WARRANTY. SHOULD THE SOFTWARE PROVE DEFECTIVE, YOU (AND NOT TELEBIT OR A TELEBIT AUTHORIZED REPRESENTATIVE) ASSUME THE ENTIRE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION. SOME STATES DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO THE ABOVE EXCLUSION MAY NOT APPLY TO YOU.

6. Limitation of Liability. UNDER NO CIRCUMSTANCES, INCLUDING NEGLIGENCE, SHALL TELEBIT BE LIABLE FOR ANY INCIDENTAL, SPECIAL OR CONSEQUENTIAL DAMAGES THAT RESULT FROM THE USE OR INABILITY TO USE THE SOFTWARE OR RELATED DOCUMENTATION, EVEN IF TELEBIT OR A TELEBIT AUTHORIZED REPRESENTATIVE HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. SOME STATES DO NOT ALLOW THE LIMITATION OF EXCLUSION OF LIABILITY FOR INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THAT THE ABOVE LIMITATION OR EXCLUSION MAY NOT APPLY TO YOU. In no event shall Telebit's total liability to you for all damages, losses, and causes of action whether in contract, tort (including negligence) or otherwise exceed the amount paid by you for the Software.

7. Controlling Law and Severability. This License shall be governed by and construed in accordance with the laws of the State of California, except that body of California law concerning conflicts of law. If for any reason a court of competent jurisdiction finds any provision of this License, or portion thereof, to be unenforceable, that provision of the License shall be enforced to the maximum extent permissible so as to effect the intent of the parties, and the remainder of this License shall continue in full force and effect.

8. Complete Agreement. This License constitutes the entire agreement between the parties with respect to the use of the Software and related documentation, and supersedes all prior or contemporaneous understandings or agreements, written or oral, regarding such subject matter.

Copyright © 1993 Telebit Corporation.

Printed in U.S.A.

How to Use This Manual

This manual is intended as a companion document to the *NetBlazer Configuration Guide*. It provides detailed information for each of the NetBlazer's commands, including an overview, syntax, options, and examples of use.

This manual is organized such that each command has its own chapter. This streamlines usability and facilitates future updating and the possible addition of new commands.

Audience

This guide is intended as a reference aid to the System Administrator. It provides specific NetBlazer command usage information to the system administrator performing NetBlazer configuration or network maintenance activities.

Related Documentation

Hardware installation of the NetBlazer products are detailed in the *NetBlazer Installation Guides*.

Once the installation of the NetBlazer has been completed, software configuration must be performed. This procedure involves establishing the IP addresses and routes and is covered in the *NetBlazer Configuration Guide*. It also covers network "fine-tuning" and management activities.

NetBlazer Training

Telebit's Customer Services group offers excellent NetBlazer training on topics such as "Introduction to Internetworking and TCP/IP," and "NetBlazer Technical Training." This comprehensive course combines easy-to-read training materials with insightful lecture and labs. For more information and class schedules, contact Telebit Education and Training Services at 1-800-TELEBIT.

Table of Contents

How to Use This Manual v
Audience v
Related Documentation v
NetBlazer Training v

Introduction INTRO-1
Typographic Conventions INTRO-1
Hierarchical Command Set INTRO-1
Common Commands INTRO-2

Alias Commands ALI-1
alias add [<alias>] [<alias-string>] ALI-1
alias delete <alias1> [<alias2> ...] ALI-2
alias list [<alias>] ALI-2

AppleTalk Commands APP-1
appletalk interface APP-1
appletalk interface add <interface> nonroute|nonseed|<net>[-<net>] <zone>... ..... APP-2
appletalk interface delete <interface> APP-3
appletalk interface disable <interface> APP-3
appletalk interface enable <interface> APP-4
appletalk interface list [<interface>] APP-4
appletalk nbplookup [-count <n>] [-interval <n>] [<object>] [:<type>] [@<zone>] APP-4
appletalk route APP-5
appletalk route add <interface> <net-range> <hops> <zone-list> APP-5
appletalk route delete <net> APP-6
appletalk route forget <interface> APP-6
appletalk route learn <interface> APP-6
appletalk route list [<interface>] APP-7
appletalk route listlocal <interface> APP-8
appletalk route lookup <net> APP-8
appletalk zonefilter APP-8
appletalk zonefilter deny <user> <zone>... ..... APP-8
appletalk zonefilter list <user> APP-9
appletalk zonefilter permit <user> <zone>... ..... APP-9

ARP Commands ARP-1
arp add <hostid> ether|token <mac-addr> ARP-1
arp delete <hostid> [<interface>] ARP-2
arp list ARP-2
arp proxy <hostid> ether|token|<interface> <mac-addr> ARP-2

Asy Commands ASY-1
asy list [<line-name>] ASY-1
asy speed <line-name> <speed> ASY-2

Bin Commands BIN-1
bin commands [terse|verbose] BIN-1
bin echo [-n] <message> BIN-1
bin edit [<filename>] BIN-2
bin eval <command> BIN-6
bin grep <string> [<filename>] BIN-6
bin more [<filename>] BIN-7
bin output [-a] <filename> BIN-8
bin passwd [<username>] BIN-8
bin settime BIN-9
bin sleep <delay> BIN-9
bin source [-v] <command-file> BIN-9
bin tcl BIN-10
bin tee [-a] <filename> BIN-11
bin time BIN-12
bin tty BIN-12
bin type <filename> BIN-12
bin uptime BIN-12
bin version BIN-12
bin wall [<message>] BIN-13
bin who [am i] BIN-13
bin write <user> [<location>] BIN-14

Boot Command BOO-1
boot [list] BOO-1
boot bootp [<delay>] BOO-1
boot normal BOO-2

Configure Command CON-1

Debug Commands DEB-1
debug alloc [<value>] DEB-1
debug args DEB-2
debug autoconfigure DEB-2
debug clear DEB-2
debug core DEB-2
debug db <address> [<count>] DEB-2
debug devs [verbose] DEB-2
debug dh <address> [<count>] DEB-2
debug display <address> [<count>] DEB-3
debug dl <address> [<count>] DEB-3
debug dump DEB-3
debug files DEB-3
debug filter DEB-3

debug gdt <start> [<end>] DEB-3
debug hardware DEB-3
debug hex <filename> DEB-4
debug hogmemory <bytes> DEB-4
debug ident <filename> DEB-4
debug idle DEB-4
debug idt <start> [<end>] DEB-4
debug inb <port> <count> DEB-4
debug irq <value> DEB-4
debug kill <pid> DEB-5
debug maplist <addr> <count> DEB-5
debug memory <command> DEB-5
debug memset <addr> <char> <length> DEB-5
debug modemcap [<modem>] DEB-5
debug noautoconfigure DEB-5
debug nocore DEB-6
debug notnclient DEB-6
debug notnserver DEB-6
debug outb <port> <value> DEB-6
debug printf <fmt> <value ...> DEB-6
debug process [<pid>] DEB-6
debug ps DEB-6
debug psignal <value> <count> DEB-7
debug remote [on|off] [en0|en1|en2] [file <filename>] [host <hostid>]
[ether <hardware-addr>] [myip <myip>] [list] [test] DEB-7
debug sessions DEB-8
debug show DEB-8
debug status [arguments] DEB-8
debug time <command> DEB-10
debug timers DEB-10
debug tnclient DEB-10
debug tnserver DEB-11
debug unmap <address> <count> DEB-11
debug watch <addr> <type> [124] [WRE] DEB-11
debug writebyte <address> <value> DEB-11
debug writelong <address> <value> DEB-11
debug writeshort <address> <value> DEB-12

Destination Commands DES-1
destination add [<dest-group>] [<short-name>] DES-1
destination delete <dest-group> [<dest-number>] DES-2
destination gid <dest-group> <group-id> DES-2
destination host <dest-group> <dest-number> <dialout>|<telnet-host>|<rlogin-host> DES-2
destination list [<dest-group>] DES-2
destination longname <dest-group> <dest-number> <long-name-string> DES-2
destination type <dest-group> <dest-number> dial | telnet | rlogin DES-3

Dialout Commands DIA-1
dialout add [<dialout>] [<phone-number>] [<characteristics-group>] [<timeout>] [<max>] [<min>] DIA-1
dialout characteristics <dialout> [#<item-number>] [<group>] DIA-2
dialout chat <dialout> <chat-script-name> DIA-2
dialout crypto <dialout> [<crypto-key>] DIA-3
dialout delete <dialout> [#<item-number>] DIA-3
dialout ditto <dialout> [#<item-number>] <phone-number> DIA-3
dialout list [<dialout>] [#<item-number>] DIA-3
dialout maxwait <dialout> <seconds> DIA-4
dialout minwait <dialout> <seconds> DIA-4
dialout password <dialout> [<remote-password>] DIA-4
dialout phone <dialout> [#<item-number>] <phone-number> DIA-4
dialout timeout <dialout> <timeout-minutes> DIA-5
dialout uname <dialout> [<remote-uname>] DIA-5

Disk Commands DIS-1
disk chmod <model+w|-w> <filename> DIS-2
disk chkdsk [verbose] [fix] [change] DIS-2
disk copy <source-file> <destination-file> DIS-3
disk delete <filename> DIS-3
disk dir [<filename.ext>] DIS-3
disk diskcopy DIS-4
disk format [system | /s] DIS-5
disk loadmodemcap DIS-5
disk rename <oldfile> <newfile> DIS-5

Files Command FIL-1

Group Command GRO-1
group add <group-name> GRO-1
group delete <group-name> GRO-1
group list [<group-name>] GRO-1
group modify <group-name> add | delete <string> GRO-2
group modify <group-name> delete #<string-number> GRO-2

History Commands HIS-1
Referencing Previous Commands HIS-2
Repeating Previous Commands HIS-2
Building New Commands from Previous Commands HIS-3
Editing Previous Commands HIS-4
history list [<number-of-lines>] HIS-5
history off HIS-6
history on [<number-of-lines>] HIS-6
history prompt HIS-6
history status HIS-6

Hostname Command HOS-1

Interface Commands INT-1
interface attach <line-name> slip|ppp <interface> [<mru>] [<ip-addr>] INT-1
interface backup <interface> [<backup-interface>] INT-3
interface delete <interface> INT-4
interface dump-mib <interface> INT-4
interface dynamic INT-4
interface dynamic add [<interface>] [<uname>] [<password>] [<crypto>]
[<chat>] [slip|ppp|ara] [<mru>] INT-5
interface dynamic auto-arp <interface> <add|delete> <ip-addr ...> INT-6
interface dynamic auto-route <interface> <add|delete> <ip-addr[/bits] ...> INT-6
interface dynamic delete <interface> INT-7
interface dynamic farside <interface> INT-7
interface dynamic framing <interface> slip|ppp INT-7
interface dynamic ipdial INT-7
interface dynamic list [<interface>] INT-8
interface dynamic pppdial [<interface>] INT-8
interface dynamic prompt <interface> [<text-string>] INT-9
interface dynamic queue-max <interface> <queue-length> INT-10
interface dynamic slipdial [<interface>] INT-10
interface dynamic timeout <interface> <timeout-seconds> INT-11
interface list [<interface>] INT-11
interface mdelay <interface> [<seconds>] [<threshold>] INT-12
interface mru [<interface>] [<mru>] INT-12
interface mtu [<interface>] [<mtu>] INT-12
interface rif <interface> [<node-address>] INT-13
interface start <interface> INT-13
interface stop <interface> INT-13
interface trace [<interface>] [<options ...>] INT-13
interface tracedump [<interface>] INT-15
interface tracelimit [<interface>] [<limit>] INT-16

IP Commands IP-1
ip address [<ip-addr>[/<bits>]] [<interface>|<group-name>|<line> [<port>]] IP-1
ip broadcast [<ip-addr> [<interface>]] IP-3
ip domain IP-3
ip domain add <server-id> IP-4
ip domain cache IP-4
ip domain delete <server-id> IP-4
ip domain flush IP-4
ip domain list IP-4
ip domain lookup <hostname> IP-4
ip domain suffix [<suffix-string>] IP-5
ip domain trace [on|off] IP-5
ip filter IP-5
ip filter add <address>[/<bits>] [<address>[/<bits>]] <interface>
[<proto> [port]] [source|destination] [in|out] permit|deny IP-6

ip filter delete <address>[/<bits>] [<address>[/<bits>]] <interface>
[<proto>] [source|destination] [in|out] IP-9
ip filter deny <address>[/<bits>] [<address>[/<bits>]] <interface>
[<proto> [port]] [source|destination] [in|out] IP-9
ip filter flush [<interface>] IP-9
ip filter list <interface> IP-9
ip filter logdeny <interface> [on|off|<count>] IP-10
ip filter lookup <address>[/<bits>] <address>[/<bits>] <interface>
[<proto> [port]] IP-10
ip filter notify [on|off] IP-11
ip filter permit <address>[/<bits>] [<address>[/<bits>]] <interface>
[<proto> [port]] [source|destination] [in|out] IP-11
ip icmp list IP-12
ip list IP-13
ip rip IP-15
ip rip accept <gateway> IP-16
ip rip add <destination> <interval> [<split-horizon>] IP-16
ip rip delete <destination> IP-16
ip rip list IP-16
ip rip netbits [<value>] IP-16
ip rip merge [on | off] IP-17
ip rip refuse <gateway> IP-17
ip rip request <gateway> [<port-number>] IP-17
ip rip timeout <interface> <seconds> IP-17
ip route IP-18
ip route add <dest-addr>[/<bits>] <interface> [<gateway>] [<metric>] IP-18
ip route addprivate <dest-addr>[/<bits>] <interface> [<gateway>] [<metric>] IP-19
ip route balance <dest-addr>[/<bits>] <interface> <interface> [<interface> ...] IP-19
ip route change <dest-addr>[/<bits>] <interface> [<gateway>] [<metric>] IP-20
ip route delete <dest-addr>[/<bits>] [<interface>] [<gateway>] IP-20
ip route flush IP-21
ip route list IP-21
ip route listbalance IP-22
ip route listlocal IP-22
ip route lookup <dest-addr> IP-22
ip route metric <dest-addr>[/<bits>] [<interface>] [<gateway>] <metric> IP-22
ip route private <dest-addr>[/<bits>] <interface> [<gateway>] IP-22
ip route public <dest-addr>[/<bits>] <interface> [<gateway>] IP-23
ip rtimer [<reassemble-time>] IP-23
ip tcp IP-23
ip tcp irtt [<int-rnd-trip-time>] IP-23
ip tcp list [<tcb>] IP-24
ip tcp mss [<max-seg-size>] IP-24
ip tcp reset <tcb> IP-24
ip tcp window [<max-window-size>] IP-25
ip ttl [<time-to-live>] IP-25
ip vj <interface> [on|off|auto] IP-25

IPX Commands IPX-1
ipx acs IPX-1
ipx acs audit [on|off] IPX-1
ipx acs deregister <user> IPX-2
ipx acs list IPX-2
ipx acs lock <line-name> [on|off] IPX-2
ipx acs start IPX-2
ipx acs stop IPX-2
ipx add <interface>|all IPX-2
ipx address [<ipx-addr> [<interface>]] IPX-3
ipx bang <ipx-addr> [<length> [<count> [<interval> [<outstanding>]]]] IPX-5
ipx delete <interface>|all IPX-5
ipx frames <interface> [ethernet_802.3|ethernet_2|token-ring|token-ring _snap] IPX-5
ipx ifconfig <interface> [<address>] [up|down] IPX-5
ipx internal-net <ipx-network-number> IPX-6
ipx iptunnel IPX-6
ipx iptunnel add <ip-addr> IPX-6
ipx iptunnel delete <ip-addr> IPX-7
ipx iptunnel flush IPX-7
ipx iptunnel list IPX-7
ipx iptunnel route <ipx-dest> <ip-gateway> [<metric>] IPX-7
ipx keepalive IPX-7
ipx keepalive list IPX-8
ipx keepalive notify [on|off] IPX-8
ipx keepalive start IPX-9
ipx keepalive stop IPX-9
ipx keepalive timeout [<max-time(minutes)>] IPX-9
ipx list IPX-9
ipx log [<facility> [<level>]] IPX-9
ipx netbios <interface> [on|off] IPX-9
ipx netstat [-AaIimrs] | [-p ipx|spx] IPX-10
ipx ping <ipx-addr> [<length> [<count> [<interval> [<outstanding>]]]] IPX-11
ipx rip IPX-11
ipx rip add <interface> [<interval (minutes)>] IPX-12
ipx rip delete <interface> IPX-12
ipx rip list IPX-12
ipx rip listen <interface>|all on|off IPX-12
ipx rip stat IPX-13
ipx route IPX-13
ipx route add <dest-addr> <interface> [<gateway> [<metric>]] IPX-13
ipx route addprivate <dest-addr> <interface> [<gateway> [<metric>]] IPX-14
ipx route bang <ipx-addr> [<length> [<count> [<interval> [<outstanding>]]]] IPX-14
ipx route delete <dest-addr> [<gateway>] IPX-14
ipx route flush IPX-14
ipx route list IPX-15
ipx route listlocal IPX-16
ipx route ping <ipx-addr> [<length> [<count> [<interval> [<outstanding>]]]] IPX-16

ipx route stat IPX-16
ipx sap IPX-16
ipx sap add <interface> [<interval (minutes)>] IPX-17
ipx sap delete <interface> IPX-17
ipx sap list IPX-17
ipx sap listen <interface>|all on|off IPX-17
ipx sap stat IPX-17
ipx sap table IPX-18
ipx sap table add [<name> [<type> [<ipx-addr>]]] IPX-18
ipx sap table delete [<name> [<type>]] IPX-19
ipx sap table list IPX-19
ipx wan IPX-20
ipx wan add <interface> IPX-20
ipx wan delete <interface> IPX-20
ipx wan list [<interface>] IPX-20
ipx wan stat IPX-20

Line Commands LIN-1
line access <line-name> [packet|all] LIN-1
line auto <line-name> [on|off|config] LIN-2
line callin <line-name> [on|off] LIN-3
line command <line-name> [<command> [<arguments>]] LIN-4
line characteristics <line-name> [<group-name-of-characteristics>] LIN-5
line ctstime <line-name> [<seconds.milliseconds>] LIN-5
line dcdtime <line-name> [<seconds.milliseconds>] LIN-6
line delay <line-name> [<milliseconds>] LIN-6
destgroup <line-name> [<destination-group>] LIN-7
line dsrtime <line-name> [<seconds.milliseconds>] LIN-7
line dtrtime <line-name> [<milliseconds>] LIN-7
line encode <line-name> [sync|async|nrzi|nrz] LIN-8
line flowcontrol <line-name> [none|auto|xon|cts] LIN-9
line idle <line-name> [minutes] LIN-10
line list [<line-name>] LIN-11
line login <line-name> [on|off] LIN-12
line modem <line-name> [<modem-name>] LIN-12
line promptdelay <line-name> [<seconds.milliseconds>] LIN-13
line request-call <line-name> [on|off] LIN-13
line reset <line-name> [dtroff|dtron] LIN-14
line speed <line-name> <speed> LIN-14
line terminal <line-name> [<terminal-type>] [-rows <rows>] [-cols <cols>] LIN-15
line test-request <line-name> LIN-15
line trace <line-name> [<options ...>] LIN-16
line v25bis <line-name> on|off LIN-17

List Command LIS-1
list ethernet LIS-2
list mem LIS-2
list pgroup LIS-2

list ps LIS-2
list slip LIS-2
list socket [<number>] LIS-2
list udp LIS-2

Logout Command LOG-1

Message Command MES-1
message hello [<message>] MES-1
message list MES-1
message login [<message>] MES-2
message shutdown [<message>] MES-2

Msglog Command MSG-1

Phonelist Commands PHO-1
phonelist delete <phonenumber...> PHO-1
phonelist deny default | <phonenumber ...> PHO-1
phonelist flush PHO-2
phonelist permit default | <phonenumber ...> PHO-2
phonelist list PHO-2
phonelist test <phonenumber> [<phonenumber> ...] PHO-2

PPP Commands PPP-1
ppp activeopen <interface> PPP-1
ppp close <interface> PPP-1
ppp list [<interface>] PPP-1
ppp options <interface> <protocol> [<option> on|off] PPP-2
ppp passiveopen <interface> PPP-2
ppp protocols <interface> [<protocol> on|off] PPP-2
ppp pool <interface> [<low-ip-addr>] [<count>] PPP-3
ppp reset <interface> PPP-4
ppp timeout <interface> [<seconds>] PPP-4
ppp trace <interface> [all lcp ipcp atcp ipxcp off] PPP-4

Reboot Command REB-1

Save Command SAV-1
save SAV-1
save local SAV-1
save remote SAV-1
save <startup> <passwd> SAV-1

Security Commands SEC-1
security login [enable|disable] SEC-1
security numberdial [enable|disable] SEC-2
security usertalk [enable|disable] SEC-2
security write [enable|disable] [user|priv|status] SEC-3

Servers Commands SER-1
servers list SER-1
servers start <server-name> [<arguments>] SER-1
servers start adsp SER-1
servers start bootpd <destination> SER-2
servers start discard [<port-number>] SER-2
servers start echo [<port-number>] SER-2
servers start finger [<port-number>] SER-2
servers start ftp [<port-number>] SER-3
servers start gdb [<port-number>] SER-3
servers start ipxsnmp [<port-number>] SER-3
servers start raw SER-3
servers start rip SER-4
servers start snmp [<port-number>] SER-4
servers start status SER-4
servers start telnet [<port-number>] SER-4
servers start watchdog <interface> [<seconds>] SER-5
servers stop <server-name> SER-5

Sessions Commands SES-1
session abort [<session-number>] SES-1
session bang <host> [<length> <count> <interval> <outstanding>] SES-1
session binary <session-number> [on|off] SES-1
session chatdial <site-name> SES-2
session close [<session-number>] SES-2
session connect <destination-name>|<destination-number> SES-2
session continue [<session-number>] SES-3
session dial <site-name>|<phone-number> SES-3
session disconnect [<session-number>] SES-3
session eol [standard | unix] SES-3
session escape [<character>] SES-3
session fg [<session-number>] SES-4
session finger [<user>] [@<host>] SES-4
session ftp [<address>] SES-5
session jobs SES-5
session kill [<session-number>] SES-5
session list SES-6
session ping <host> [<length> <count> <interval> <outstanding>] SES-6
session rlogin [-b|-8] [-noesc] [-l username] [<address>] [<port>] SES-7
session send <session-number> ao|ayt|brk|ec|el|esc|ga|ip|nop|synch SES-8
session talk [<line-name>] SES-8
session telnet [-b|-8|-raw] [-noesc] [<address>] [<port>] SES-9
session terminal [<terminal-type>] [-rows <rows>] [-cols <cols>] SES-9
session tftp [<address>] SES-10

Setconsole Command SET-1

Shutdown Command SHU-1

SNMP Commands SNM-1
Community Groups and Security SNM-1
Trap Community SNM-1
Privileges SNM-2
The Telebit Enterprise MIB SNM-2
SNMP over IPX SNM-3
snmp authentication on|off SNM-3
snmp community SNM-3
snmp community add <community> [<ro>|<rw>] SNM-3
snmp community delete <community> SNM-4
snmp community host <community-name> add|delete <net-type><hostid> SNM-4
snmp community list [<community-name>] SNM-4
snmp enterprise [disable|default|enable [ro|rw]] SNM-4
snmp list SNM-5
snmp startup on|off SNM-6
snmp sysContact <sysContact-string> SNM-6
snmp sysLocation [<location>] SNM-6
snmp sysName [<name>] SNM-6

Syslog Commands SYS-1
syslog buffer off | on [<level>|<=level>] | list [<file name>] | size <sizeK> SYS-2
syslog console off | on [<level>|<=level>] SYS-3
syslog host off | on [<loghostname>] [<facility>] SYS-3
syslog interval flush | off | on [<interval in minutes>] SYS-4
syslog list SYS-4
syslog session off | on [<level>|<=level>] SYS-5

User Commands USE-1
Types of User Records USE-1
user add [<userid>] USE-2
user changeable [on|off] USE-2
user command <userid> [<command> ...] USE-3
user crypto <userid> [<crypto-key>] USE-3
user delete [<userid>] USE-4
user destgroup <userid> <destination-group> USE-4
user domainlookup [on|off] USE-4
user homezone <userid> [<zone>] USE-4
user list [<userid>] USE-5
user load [<host:filename>] USE-5
user local <userid> USE-5
user password <userid> [<password>] USE-6
user privilege <userid> [[-]conf] [[-]stat] USE-6
user remote [<host>:<filename>] USE-6
user restrict <userid> [[-]dial] [[-]telnet] USE-7
user timelimit <userid> [<minutes>|unlimited] USE-8
user userdomain [<domainname>] USE-8

Introduction

This manual provides a complete description of all the commands used by the NetBlazer. Each command description shows the command format followed by an explanation of the command and its associated parameters.

Typographic Conventions

The following typographic conventions are used in this manual.

keyword — Key words shown in bold type are NetBlazer commands. You usually need to enter only enough of the key word to uniquely distinguish it from other key words used by the NetBlazer. For example, entering **host** is equivalent to entering **hostname**.

<parameter-name> — Parameter names enclosed in angle brackets serve as placeholders for information that you must supply. Substitute an appropriate value for the named parameter.

[optional parameters] — Parameters enclosed in square brackets are optional. A series of optional parameters, such as [<param-1>] [<param-2>] [<param-3>], must be entered in sequence. In other words, <param-1> must be entered before <param-2> and <param-2> must be entered before <param-3>.

choice1 | choice2 — Vertical bars indicate a choice between two or more items. You must choose one of the items unless all of the items are enclosed in square brackets and are therefore optional, for example, [choice1 | choice2 | choice3].

Hierarchical Command Set

As shown in Figures INTRO-1, INTRO-2, INTRO-3, INTRO-4, and INTRO-5, the NetBlazer commands are arranged in a tree-structured hierarchy, starting at the top command level and working down through several subcommand levels. Table INTRO-1 provides a summary of the commands by functional group.

When you enter most commands, the NetBlazer first checks the current command level for the command. If it is a valid command for that level, the NetBlazer invokes the function associated with the command at that level. If not, the NetBlazer searches the command tree for the command. For example, if you are currently at the Configure User command level, entering **list** displays a list of users, whereas entering **route list** displays the list of IP routes without leaving the current command level.

To go to a new command level, enter the name of the new command level. Usually, you need to enter only enough of the command name to uniquely distinguish it from other commands. For example, to get to the Configure IP Route command level from anywhere in the command tree, enter either **route** or **ro** at the command prompt.

For the **ipx route** command, enter **ipx route** or **ipx ro** to get to the Configure IPX Route command level. For the **ipx rip** command, enter **ipx rip** to get to the Configure IPX RIP command level. Alternatively, you can enter **ipx** to get to the Configure IPX command level. When you are at that level, you need to enter only the command itself, such as **rip** or **route**.

For the **appletalk route** command, type **appletalk route** to get to the Configure AppleTalk Route command level. For the **appletalk interface** command, enter **appletalk interface** to get to the Configure AppleTalk Interface command level. Alternatively, you can enter **apple** or **appletalk** to get to the Configure AppleTalk command level. When you are at that level, you need to enter only the command itself, such as **interface** or **route**.

If a command is not a valid command for a level and the NetBlazer cannot locate the command at another level, the NetBlazer displays the message **Unknown command; type "?" for list of local commands**. If you see this message, move to the command level you want by using the **up** or **top** command (see the next section, Common Commands).

Common Commands

Three commands are common to every level in the command tree: **top**, **up**, and **?**.

The **?** command displays a list of commands at the current command level. If you enter a command and a space before the **?**, the syntax for that command is displayed.

The **up** command takes you up to the next higher level in the command tree. For example, if you are currently at the Configure IP command level, enter an **up** command to move up one level to the Configure command level.

The **top** command takes you to the top level of the command tree.

This manual provides a description of each NetBlazer command arranged in alphabetical order by submenu. For example, the **ipx** commands at the Configure IPX command level are described under the heading IPX Commands, and the **ipx route** commands at the Configure IPX Route command level are described under the IPX Route heading.

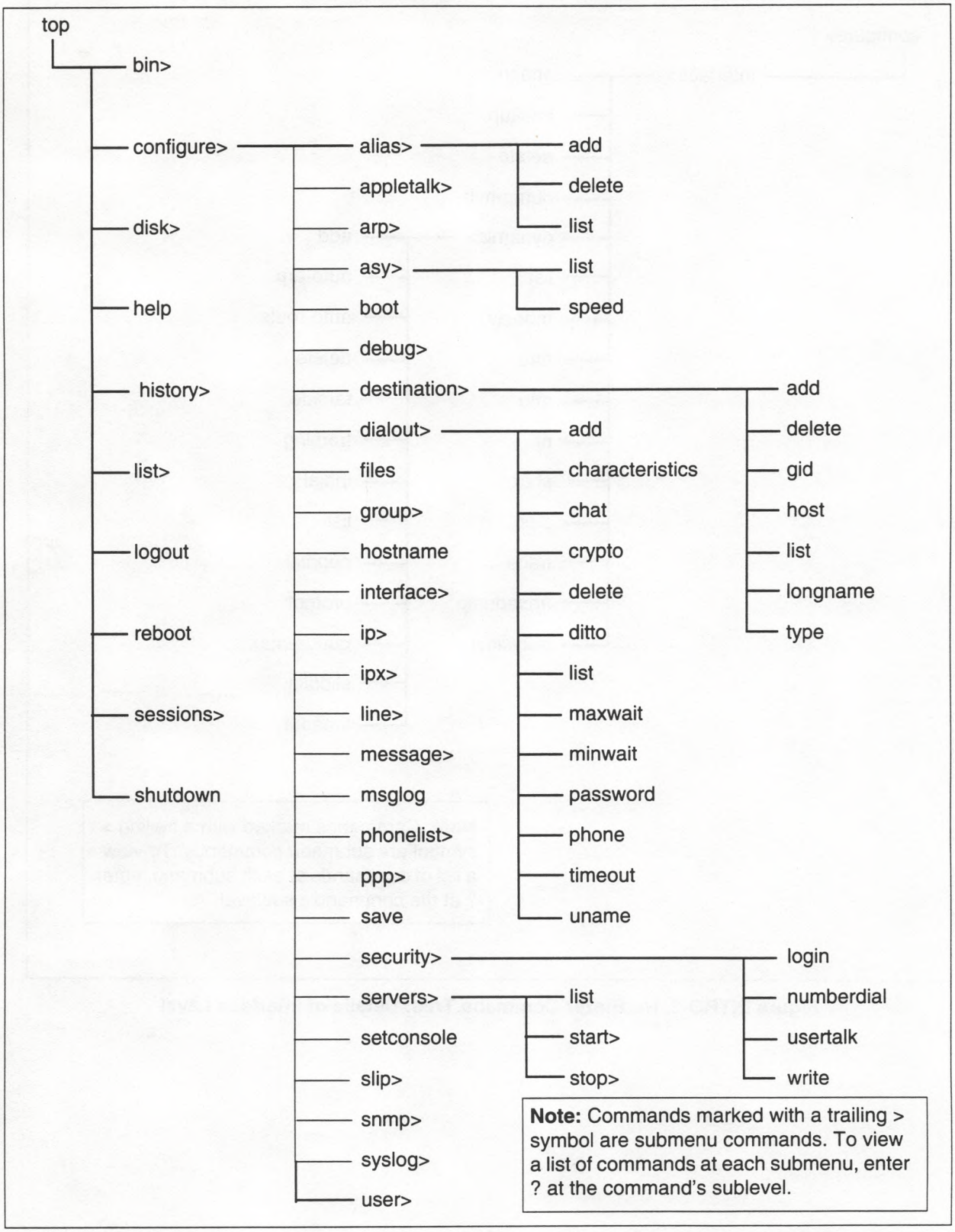

Figure INTRO-1. NetBlazer Command Tree: Details of Alias, Asy, Destination, Dialout, Security, and Servers Levels

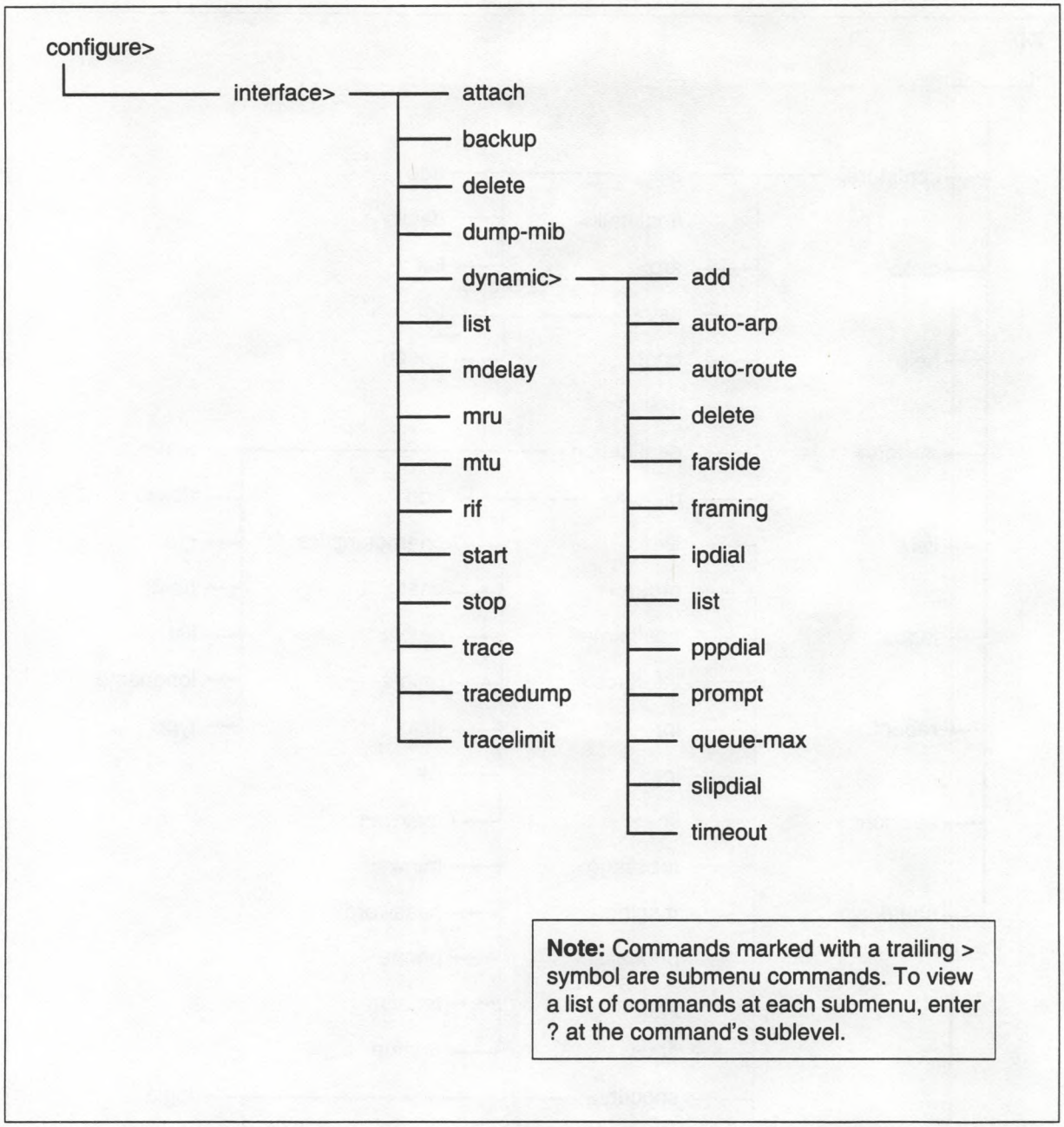

Figure INTRO-2. NetBlazer Command Tree: Details of Interface Level

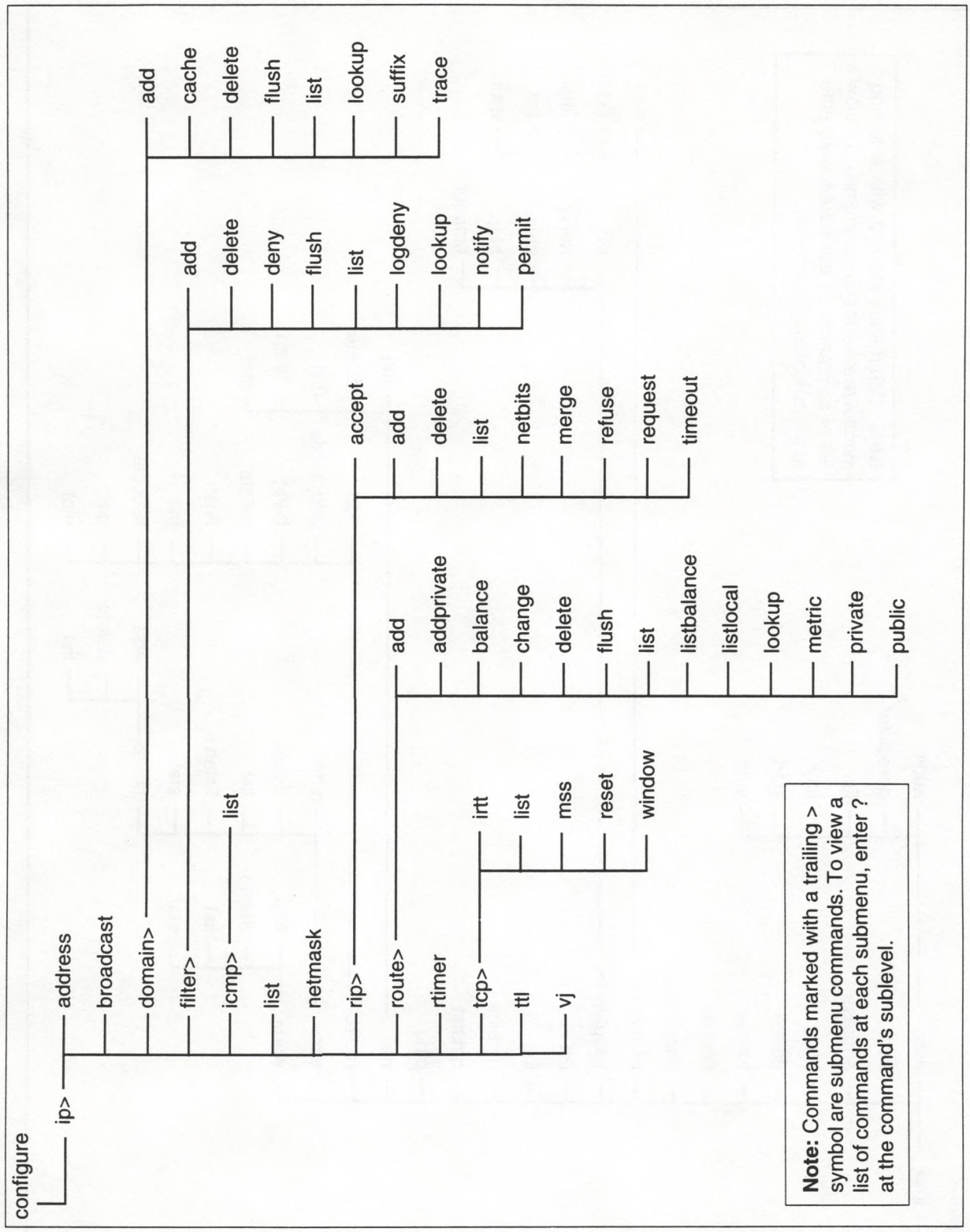

Figure INTRO-3. NetBlazer Command Tree: Detail of IP Command

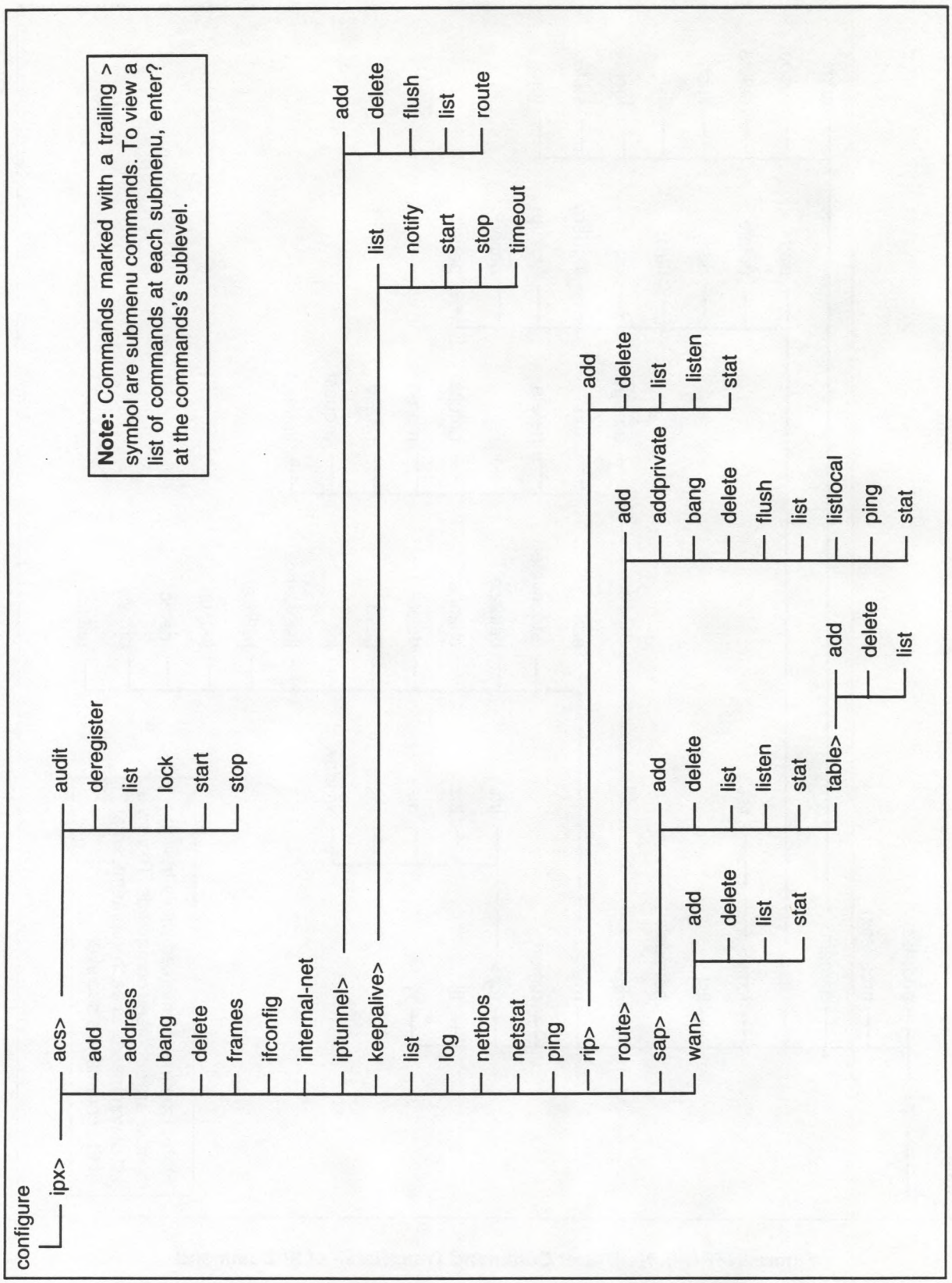

Figure INTRO-4. NetBlazer Command Tree: Details of IPX Level

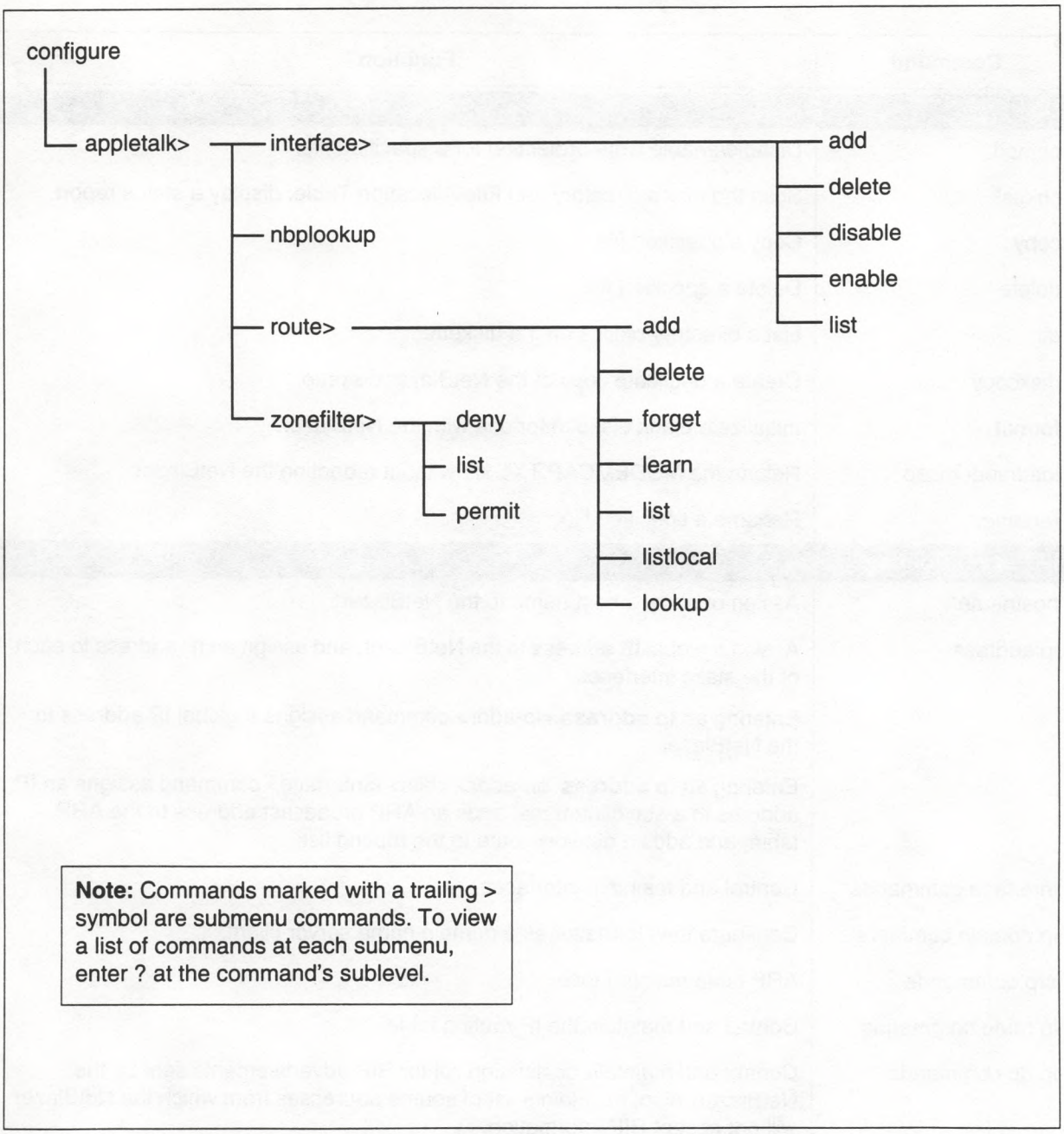

Figure INTRO-5. NetBlazer Command Tree: Details of AppleTalk Level

Table INTRO-1. Command Summary

Command	Function
Disk	
chmod	Disable/enable write protection for a specified file.
chkdsk	Scan the disk's directory and File Allocation Table; display a status report.
copy	Copy a specified file.
delete	Delete a specified file.
dir	List a directory of files on the diskette.
diskcopy	Create a duplicate copy of the NetBlazer diskette.
format	Initialize a blank diskette for use with the NetBlazer.
loadmodemcap	Reload the MODEMCAP.TXT file without rebooting the NetBlazer.
rename	Rename a specified file.
IP Packet Routing	
hostname	Assign or show a host name to the NetBlazer.
ip address	Assign a global IP address to the NetBlazer, and assign an IP address to each of the static interfaces. Entering an **ip address** <ip-addr> command assigns a global IP address to the NetBlazer. Entering an **ip address** <ip-addr>/<bits> <interface> command assigns an IP address to a static interface, adds an ARP broadcast address to the ARP table, and adds a network route to the routing list.
interface commands	Control and maintain interfaces.
ip domain commands	Configure the NetBlazer as a domain name server client.
arp commands	ARP table maintenance.
ip route commands	Control and maintain the IP routing table.
ip rip commands	Control and maintain destination list for RIP advertisements sent by the NetBlazer. Also, maintain a list of source addresses from which the NetBlazer will not accept RIP information.
ip filter commands	Maintain a list of IP packet filters used to restrict IP packet routing through a particular interface.
IPX Packet Routing	
hostname	Assign or show a host name to the NetBlazer.
interface commands	Control and maintain interfaces.
ipx add	Prepare an existing NetBlazer interface for use by IPX.

Table INTRO-1. Command Summary (Continued)

Command	Function
ipx address	Assign a global IPX address to the NetBlazer, and assign an IPX address to each of the static interfaces. Entering an **ipx address** <ip-addr> command assigns a global IPX address to the NetBlazer. Entering an **ipx address** [<ipx-addr> [<interface>]] command assigns an IPX address to a static interface.
ipx iptunnel commands	Configure and manage IP tunneling. IP tunneling is a way of carrying an IPX datagram in an IP packet so IPX networks can be connected across internetworks that carry only IP traffic.
ipx keepalive commands	Manage IPX keep-alive messages and responses.
ipx route commands	Control and maintain the IPX routing table.
ipx rip commands	Control and maintain destination list for RIP advertisements sent by the NetBlazer. Also, maintain a list of source addresses from which the NetBlazer will not accept RIP information.
ipx sap commands	Control and maintain list of server advertisements sent and received by the NetBlazer.
ipx wan commands	Enable or disable Novell's Wide Area Network (WAN) protocol support.
AppleTalk Packet Routing	
hostname	Assign or show a host name to the NetBlazer.
appletalk interface commands	Control and maintain AppleTalk interfaces
appletalk nbplookup	Enable a search for named devices on the AppleTalk network.
appletalk remote commands	Set up and maintain AppleTalk Remote Access (ARA) users.
appletalk route commands	Control and maintain the AppleTalk routing table.
appletalk zonefilter commands	Lock ARA users out of specified zones.
IP Synchronous Interface	
interface commands	Synchronous interface (syn0) control and maintenance.
ip route commands	Control routing IP packets through the syn0 interface.
ppp commands	Control the point-to-point framing protocol used to transfer packets over the synchronous link.

Table INTRO-1. Command Summary (Continued)

Command	Function
Dedicated SLIP or PPP Link	
interface commands	Attach a packet mode interface to an asynchronous line.
ppp commands	Control the point-to-point framing protocol used to transfer packets over the dedicated PPP link.
IP Dynamic (Dial-Up) Interface	
ipdial	Start a dialog that steps you through the process of creating a dynamic dial-up interface to a remote site. The NetBlazer uses information obtained from ipdial to create: • a dynamic interface record for the remote site. • one or more dial-out records used to call the remote site. • a user record to authenticate a dial-in connection from the remote site. • routes for the remote site.
dynamic commands	Set up and maintain dynamic interfaces; used to route IP packets over the public telephone network.
ip route commands	Maintain routes to remote networks through the dynamic interfaces.
dialout commands	Maintain a list of dial-out destinations used to establish a connection with a remote site.
user commands	Maintain a list of users authorized to log into the NetBlazer.
Users and Destination Groups	
user commands	Set up and maintain list of authorized users.
destination commands	Create destination groups, which allows authorized user to easily start a telnet or dial-out session with a predefined destination.
Sessions	
abort	Abort an FTP data transfer operation.
bang	Equivalent to the ping command except, instead of printing a line of statistics for each response, only a single ! character is displayed.
binary	Turn binary mode on or off for a specified telnet session.
close	End a telnet session with a graceful, negotiated termination.
connect	Start a telnet or dialout session t using a predefined destination name.
continue and fg	Reactivate a session after it has been suspended.
dial	Establish a character mode connection with a remote site specified by a dial-out record.
disconnect	Terminate a character mode dial-out connection.
eol	UNIX end-of-line marker. Functionally equivalent to a Return or Enter key-stroke. Used when a remote UNIX host does not respond to a normal Return.

Table INTRO-1. Command Summary (Continued)

Command	Function
escape	Display or change the keystroke used in a escape sequence to interrupt a session to get back to the command prompt.
finger	Get information about a specific user on the network.
ftp	Transfer files between the NetBlazer and a remote host that supports FTP.
tftp	Transfer files for remote configuration between the NetBlazer and a remote host that supports TFTP.
list and jobs	Display a list of sessions currently active on the NetBlazer.
kill	Force an abrupt termination of a session.
ping	Send ICMP echo requests to a specified host, then measure the time taken for the host to respond.
rlogin	Set up an interactive session on remote host using current NetBlazer user ID as remote host login ID.
send	Issue a predefined character sequence to a remote host, such as an escape sequence.
talk	For debugging purposes. Connect to an asynchronous line and issue modem commands to an attached modem.
telnet	Establish an interactive session with an TCP/IP host.
terminal	Set terminal information for use by a UNIX host.
Diagnostic/Network Management	
snmp commands	Obtain statistical and other network management information.
syslog commands	Send NetBlazer activity information to a specified UNIX host, to a terminal, to a buffer in memory, or to the console.
icmp	Display a list of error and control statistics maintained by the NetBlazer.
tcp commands	Set up and manage specific TCP (Transmission Control Protocol) parameters used by the NetBlazers.
list commands	Display various NetBlazer status information.
debug commands	A suite of network and NetBlazer troubleshooting facilities.
Servers	
adsp	Start the AppleTalk Data Stream Protocol (ADSP) server. ADSP is a sessions-level protocol in the AppleTalk protocol stack. ADSP allows an interactive session with a remote host. ADSP supports terminal emulators that perform functions similar to IP's TELNET.
bootpd	Start the bootp server.
discard	Accept and acknowledge data received on TCP port 9. Data is discarded.
echo	Accept and acknowledge TCP segments received at port 7. Data is echoed back to the original source.

Table INTRO-1. Command Summary (Continued)

Command	Function
finger	Allow other hosts to query the NetBlazer for information about NetBlazer users using the FINGER protocol.
ftp	Enable use of FTP (File Transfer Protocol) to transfer new software and configuration files to the NetBlazer from a remote host.
raw	Allow a raw TCP connection from a host to the modem or other asynchronous device attached to a NetBlazer line.
rip	Allow the NetBlazer to learn IP routes to other networks from other routers on the network.
snmp	Perform network management tasks.
ipxsnmp	Perform IPX network management tasks.
telnet	Allow users to log into the NetBlazer from a TCP/IP host.
watchdog	Monitor an interface that is using PPP.
Productivity Tools	
alias	Set up and maintain command aliases.
grep	Search a file for a specified string. Can also pipe output of a list to grep.
history commands	Maintain and display a list of the last 20 commands entered into the NetBlazer. Provides a "shorthand" method of entering commands based on the history list.
message commands	Enter and maintain messages displayed at start-up, login, and shutdown.
more	Display the contents of NetBlazer files on diskette a page (24 lines) at a time. The output of a list can also be piped to more.
source	Run NetBlazer commands stored in a file.
who	List all users currently logged onto the NetBlazer.
write	Write a message to a user currently logged onto the NetBlazer.
wall	Write to all users currently logged onto the NetBlazer.

Alias Commands

The **alias** command takes you to the Configure Alias command level. To view a list of the commands available at this level, type a question mark (?) at the Configure Alias command level.

```
NB:Top> alias
NB:Top>Configure>Alias> ?
Available commands:
add             delete          list            top>
up>             ?
```

Commands available at this level maintain a list of aliases that are used as abbreviations for standard NetBlazer commands.

Caution: *Do not use an existing command word as an alias. Doing so renders the command inaccessible.*

alias add [<alias>] [<alias-string>]

The **alias add** command assigns a command string to an alias key word. The <alias> parameter is an abbreviation used to invoke the NetBlazer command specified by the <alias-string> parameter.

The following example shows how the **update** alias was created at Telebit:

```
NB:Top> alias
NB:Top>Configure>Alias> add update ftp 143.191.20.1
NB:Top>Configure>Alias> list
Alias                   Expansion
update                  ftp 143.191.20.1
```

The **update** alias shown above is a shorthand way to start an FTP session at 143.191.20.1 which is used to download the latest NetBlazer software.

If you enter an **alias add** command without any parameters, the system prompts you for the alias key word and its associated command string.

```
NB:Top>Configure>Alias> add
What command word would you like to alias? wt
What would you like to alias it to? who;time
```

The command string assigned to an alias can be a single command or multiple commands separated by a semicolon.

Notes to previous users

*The **login_<user>** alias, which was used in earlier versions of the NetBlazer software to assign users a login command, will not be supported in future releases.*

*The NetBlazer now uses **user command** to automatically invoke a command, such as telnet, when a user logs in to the NetBlazer. Please replace existing login_<user> aliases with user login commands. See the **user command** description for information on assigning login commands to users.*

alias delete <alias1> [<alias2> ...]

The **alias delete** command removes one or more alias key words and their associated command strings from the list of valid aliases.

```
NB:Top>Configure>Alias> list
Alias                   Expansion
bye                     logout
quit                    logout
update                  ftp 143.191.20.1
NB:Top>Configure>Alias> delete quit bye
NB:Top>Configure>Alias> list
Alias                   Expansion
update                  ftp 143.191.20.1
```

alias list [<alias>]

The **alias list** command displays the specified alias key words and their associated command string. If this command is entered without a <alias> parameter, all aliases are displayed.

```
NB:Top>Configure>Alias> list
Alias                   Expansion
bye                     logout
quit                    logout
update                  ftp 143.191.20.1
NB:Top>Configure>Alias> list quit bye
Alias                   Expansion
           quit         logout
            bye         logout
```

AppleTalk Commands

The AppleTalk commands reside in two main submenus under the NetBlazer Configuration menu: **configure appletalk** and **configure user**. Commands associated with establishing and maintaining users under the **configure user** submenu include **add**, **delete**, **homezone**, **password**, **remote**, and **timelimit**. Refer to the User section of this manual for details on these commands.

An AppleTalk Data Stream Protocol (ADSP) server is provided in the NetBlazer software to support terminal emulation (dial-out) applications. Activation (or deactivation) of this server is performed through the Servers menu.

Commands under the AppleTalk submenu relate to setting up AppleTalk interfaces, routes, zones, and zone filtering. To view a list of the commands available at this level, type a question mark (?) at the Configure AppleTalk command level.

```
NB:Top>Configure>AppleTalk> ?
Available commands:
interface>      nbplookup       route>          top>
up>             zonefilter>     ?
```

appletalk interface

The **appletalk interface** command takes you to the Configure AppleTalk Interface command level. To view a list of commands at this level, type a question mark (?) at the Configure AppleTalk Interface level.

```
NB:Top>Configure>AppleTalk>Interface> ?
Available commands:
add             delete          disable         enable
list            top>            up>             ?
```

Commands at this level are used to configure and control the AppleTalk interfaces used by the NetBlazer.

The NetBlazer uses three types of AppleTalk interfaces: LAN, dynamic and dedicated.

LAN interfaces are automatically created for each Ethernet and Token Ring board installed in the NetBlazer. Before using these interfaces, you must configure the interfaces as described in the *NetBlazer Configuration Guide*.

Dynamic interfaces provide dial-up links to route AppleTalk packets over the public switched telephone network. Refer to the **interface dynamic** command descriptions for instructions on setting and maintaining dynamic interfaces.

Dedicated interfaces route packets over dedicated point-to-point serial links. Use the **interface attach** command to attach a dedicated interface to a synchronous or asynchronous serial line.

You can start a **watchdog** server to monitor the state of a dedicated interface. If the dedicated interface link fails, the NetBlazer automatically reroutes the link's traffic through a dynamic backup interface. A backup interface is assigned to a dedicated interface through the **interface backup** command.

appletalk interface add <interface> nonroute|nonseed|<net>[-<net>] <zone>...

This command is used to set up an AppleTalk Ethernet interface, or a PPP interface.

```
NB:Top>Configure>AppleTalk>Interface> add ?
Usage: add <interface> <net#>[-<net#>] <zone>... # seed routing
       add <interface> nonseed                   # nonseed routing
       add <interface> nonroute                  # no routing
       add <interface>                           # PPP interfaces
 ## add an AppleTalk interface ##
```

Notes:

1. *The first interface added, whether en0, en1, or en02, becomes the default AppleTalk interface. In addition, the first zone specified on that interface becomes the home zone. Although as many zone names as desired may be assigned to that interface, the NetBlazer will logically reside in the first (home) zone.*

2. *The <net#>[-<net#>], nonroute, nonseed, and <zone> parameters cannot be specified for dynamic interfaces. Zones may be specified only for seed interfaces.*

<net>[-<net>]

Either a specific network, or a range of network numbers can be specified in the <net#> -<net#> parameter. A single number, or two numbers separated with a hyphen (-) can be used. The range can be from 1 to 65,279 (FEFF hex). Note that the first network number must be less than or equal to the second number.

nonroute

The nonroute option may be used when it is desired to have the NetBlazer appear only as a node on the network. This is desirable in situations when the NetBlazer is being used only as an ARA server, for example. Note that nonroute is mutually exclusive with nonseed.

nonseed

The nonseed option may be used to acquire route information from another router on the LAN through RTMP, provided there is a seed AppleTalk router on the network. See also the AppleTalk Configuration section in the NetBlazer Configuration Guide.

<zone>

<zone> is a list of strings that are the names of the zones that apply to the particular interface.

Note: *If spaces or special characters are used in the zone, it is necessary to surround them with double quotes in order for the NetBlazer to treat the string as one argument.*

appletalk interface delete <interface>

The **delete** command allows a specified AppleTalk interface to be deleted.

Note: *If the default appletalk interface is deleted using this command, a warning message is displayed to notify the user that some NetBlazer AppleTalk services (including ARA) will be unavailable until another AppleTalk interface is added or enabled (thus becoming the new default interface).*

```
NB:Top>Configure>AppleTalk>Interface> disable en0
WARNING: If the default AppleTalk interface ('en0') is disabled,
some network services (including AppleTalk Remote Access) will be
unavailable until a new default port has been designated.
Are you sure you want to do this (y/n) [n]? y
```

appletalk interface disable <interface>

The **disable** command allows a specified interface to be taken down without having to re-enter any configuration information. Its counterpart is the **enable** command.

Notes:

1. *This command is operative only on AppleTalk. For example, entering **appletalk interface disable en0** shuts off only AppleTalk activity on that interface. IP and IPX activity are unaffected.*
2. *If the default AppleTalk interface is disabled using this command, a warning message is displayed to notify the user that some NetBlazer AppleTalk services (including ARA) will be unavailable until the interface is enabled.*

appletalk interface enable <interface>

The **enable** command allows a specified interface to be brought back up (after disabling it) without having to re-enter any configuration information. Its counterpart is the disable command.

Note: *This command is operative only on AppleTalk. For example, entering* ***appletalk interface enable en0*** *resumes AppleTalk activity on that interface. IP and IPX activity are unaffected.*

appletalk interface list [<interface>]

The **list** commands lists all AppleTalk interfaces and their configuration.

```
NB:Top>Configure>AppleTalk>Interface> list
port #  name      network range  zones
------  --------  -------------  ----------------------------------------
*   0   en0         100          Marketing Finance Admin
```

The interface parameter is optional. If no interface is specified, the information listed is for all interfaces.

appletalk nbplookup [-count <n>] [-interval <n>] [<object>] [:<type>] [@<zone>]

The **nbplookup** command is provided as a convenience to the system administrator when performing initial NetBlazer AppleTalk configuration. It enables a search for named devices on the AppleTalk network. For example, it can be used to locate all the file servers within a particular zone.

[-count <n>] [-interval <n>]

The [-count <n>] parameter is the number of npb requests to send, while the [-interval <n>] parameter is time to delay between requests, in seconds. The default value for count is 3 requests, and 1 second for interval. Note that because the **nbplookup** command cannot be interrupted, the maximum interval value is one minute.

[<object>][:<type>][@<zone>]

The [<object>][:<type>][@<zone>] subfields are actually are a single parameter and therefore should not be separated with spaces. In cases where the [<object>][:<type>][@<zone>] name actually contains a space, the whole argument must be enclosed in double quotes.

Note: *If spaces or special characters are used in the object name, it is necessary to surround them with double quotes in order for the NetBlazer to see the string as one argument.*

The [<object>] portion of the argument is the name of a specific entity, while [:<type>] is the general class of devices, such as AFPServer, workstation, SNMP agent, and so on. For example, on a simple network that contains three AFP fileservers, entering nbplookup :afpserver would return information on all three file servers. Entering nbplookup poobear:afpserver, would return information only on the AFPServer named poobear.

If either of the [<object>][:<type>] subfields are omitted, the default value is displayed as an equal (=) sign. The equal sign can also be explicitly specified to indicate a wildcard (AppleTalk standard convention).

When the <zone> field is omitted, the NetBlazer's default zone is assumed.

Examples:

nbplookup :AFPServer	Finds all fileservers in the default zone.
nbplookup :LaserWriter	Finds all LaserWriters in the default zone.

appletalk route

AppleTalk routes are dynamically discovered as a function of AppleTalk RTMP (Route Table Management Protocol). However, in order to perform on-demand dialup through the use of PPP, it is necessary to create static routes. The **add** and **delete** commands are used for this purpose.

The **learn** and the **forget** commands are provided as a convenience when dialing into large AppleTalk networks where it is not practical to manually add, and then delete a large number of routes.

```
NB:Top>Configure>AppleTalk>Route> ?
Available commands:
add             delete          forget          learn
list            listlocal       lookup          top>
up>             ?
```

appletalk route add <interface> <net-range> <hops> <zone-list>

The **appletalk route add** command is used to add a static route. The use of the **learn** command is recommended in adding routes.

<interface>

This parameter is the interface you wish to use, and may be static or dynamic PPP interfaces.

<net-range>

Either a specific network, or a range of network numbers can be specified in the <net-range> parameter. A single number, or two numbers separated with a hyphen (-) can be used. The range can be from 1 to 65,279 (FEFF hex). Note that the first network number must be less than or equal to the second number.

<hops>

Hops are the number of networks between the NetBlazer and the final route destination. The network the NetBlazer is on counts as number one.

<zone-list>

<zone> is a list of strings that are the names of the zones that apply to the particular interface.

appletalk route delete <net>

The **appletalk route delete** command is used to remove a specified static or learned route.

appletalk route forget <interface>

The **appletalk route forget** command is used to delete all learned routes for a specified interface.

The **appletalk route forget** command is typically followed by use of the **learn** command to acquire new route information, and then a **save**.

appletalk route learn <interface>

The **appletalk route learn** command, when given the name of the PPP interface, will automatically establish the connection (if not already open), and learn what networks exist on the far side.

This information obtained from the network is then automatically added to the routing table and specifically marked as having been learned. Learned route information is distinct from RTMP acquired and manually added routes, thereby allowing learned routes to be automatically removed with the **forget** command.

appletalk route list [<interface>]

The **list** command lists all routes including learned, dynamic (RTMP), and manually entered routes for a specified interface. If no interface is specified, routes for all interfaces are listed.

```
NB:Top>Configure>AppleTalk>Route> list
1 routes
 net range   state    interface  hops  next router  zones
-----------  -------  ---------  ----  -----------  -----
  100        good     en0          0                Admin Finance Marketing
```

net range

Shows the networks to which the NetBlazer provides a route.

type

Shows how the route information was obtained (static, learned, or RTMP).

state

Shows Static, Learned and RTMP discovered routes.

In the case of RTMP learned routes, table entries age and eventually time out. The State field shows the current status of the entry. Relative age is indicated by:

good, suspect, and bad

interface

Shows the interface through which packets addressed to the network indicated will be routed.

hops

Shows hop count (up to 15).

next router

Show the network and node number of the next router. If the hop count is greater than 0, packets must be forwarded through another router to reach the destination.

zones

Shows the list of zones that are available on the network.

appletalk route listlocal <interface>

This command is used to list only static and learned routes for a specified interface. If no interface is specified, all interfaces are shown.

```
NB:Top>Configure>AppleTalk>Route> listlocal en0
3 routes
 net range   state    interface  hops  next router  zones
-----------  -------  ---------  ----  -----------  --------------------
1000         learned  Finance en0   1               Payables Receivables
2000         learned  Finance en0   1               G/L Payroll
3000         learned  Finance en0   1               CorpCntrlr IntlCntlr
```

appletalk route lookup <net>

The **lookup** command is used to identify the route for a specified network.

appletalk zonefilter

AppleTalk zonefilter facilities are used to control remote user access to zones. Filter configurations are available on a per user basis.

Once ARA users have been initially configured, they have access to all AppleTalk zones. The filter **deny** command is used to prevent an ARA user from having access to specified zone(s), while all other zones are permitted. Conversely, the **permit** command prevents access to all zones except those explicitly specified.

This command is particularly useful when you wish to have remote guest access to your network, but only to a designated zone or zones.

```
NetBlazer:Top>Configure>AppleTalk>Zonefilter> ?
Available commands:
deny            list            permit          top>
up>             ?
NetBlazer:Top>Configure>AppleTalk>Zonefilter>
```

appletalk zonefilter deny <user> <zone>...

The **deny** command prevents a remote user from having access to one or more specified zones. Initially, neither **deny** nor **permit** are active. These commands are mutually exclusive. Specifying, for example,

```
NB:Top>Configure>AppleTalk>Zonefilter> deny Fred Accounting Shipping
```

user Fred is prevented from accessing either of those zones, while allowed access to all others.

Note: *The permit and deny zone list for each user can be modified by specifying the desired zone or zones to be permitted or denied.*

For example, if

```
NB:Top>Configure>AppleTalk>Zonefilter> deny Fred Accounting Shipping
```

is in effect, then a

```
NB:Top>Configure>AppleTalk>Zonefilter> permit Fred Shipping
```

is issued, zone Shipping is effectively removed from the deny list maintained for user Fred.

appletalk zonefilter list <user>

This command lists the zones to which a remote user is permitted or denied access. Depending on which command was used (**permit** or **deny**), a list of all permitted zones, or all denied zones is displayed (never both).

appletalk zonefilter permit <user> <zone>...

The **permit** command is used to assign a remote user access to one or more logical zones. Initially, neither **permit** nor **deny** are active. Specifying, for example, **permit Fred Accounting Shipping**, user Fred is granted access to those zones and prevented access from all others.

Note: *The permit and deny zone list for each user can be modified by specifying the desired zone or zones to be permitted or denied. For example, if **deny Fred Accounting Shipping** is in effect, issuing **permit Fred Shipping** effectively removes zone Shipping from the deny list maintained for Fred.*

ARP Commands

The **arp** command takes you to the Configure ARP command level. To view a list of the commands available at this level, type a question mark (?) at the Configure ARP command level.

```
NB:Top> arp
NB:Top>Configure>ARP> ?
Available commands:
add             delete          list            proxy
top>            up>             ?
```

The ARP (Address Resolution Protocol) commands are used to maintain a cache of IP addresses and corresponding Ethernet or Token Ring addresses, which allows the NetBlazer to find the Ethernet or Token Ring address of a host on the same network when given only the destination host's IP address.

Note: *When RIP is used to dynamically maintain routing information, the broadcast addresses of each Ethernet network connected directly to the NetBlazer must be added to the ARP list. The preferred method for adding broadcast addresses to the ARP list is to assign an IP address to the Ethernet interface using a subnet bit mask. Refer to the* ***ip address*** *command description for additional information.*

arp add <hostid> ether|token <mac-addr>

The **arp add** command adds an IP address and its associated Ethernet or Token Ring address to the ARP list. This command can be used to provide support for systems that do not support ARP or to work around a routing problem.

The <hostid> may be a domain name or IP address of a specific host, or the broadcast IP address for a network.

The ether <mac-addr> or token <mac-addr> parameter specifies the physical address that is mapped to the <hostid>.

One case where you may need to add an ARP entry is if your network requires an X.0 broadcast address. The following example shows you how to add this type of broadcast address to the ARP list.

```
NB:Top>Configure>ARP> add 143.191.63.0 ether ff:ff:ff:ff:ff:ff
NB:Top>Configure>ARP> list
received 67 badtype 0 bogus addr 0 reqst in 64 replies 3 reqst out 7
IP addr         Type  Interface   Time Q Addr
143.191.63.255  ether any         0      ff:ff:ff:ff:ff:ff
143.191.63.0    ether any         0      ff:ff:ff:ff:ff:ff
```

arp delete <hostid> [<interface>]

The **arp delete** command removes an entry in the ARP list. This command may be used when an IP address assigned to an Ethernet address is no longer valid, which may occur when an Ethernet board is changed or the IP address has been assigned to a different host or router.

Note that the <interface> parameter is only required when deleting a proxy ARP entry from the list.

arp list

The **arp list** command displays the current ARP status and a list of related IP and Ethernet addresses. The ARP list contains two types of entries: static entries, which are manually entered by **arp add** commands, and dynamic entries, which are automatically added to the list in response to an ARP request.

The following ARP list shows that IP address 143.191.11.252 is mapped to Ethernet address 00:80:ad:02:62:6b and that the entry will be discarded in 893 seconds. The first two entries are static entries, which only can be removed via an **arp delete** command.

```
NB:Top>Configure>ARP> list
received 70 badtype 0 bogus addr 0 reqst in 67 replies 3 reqst out 7
IP addr         Type  Interface   Time Q Addr
143.191.63.255  ether any         0      ff:ff:ff:ff:ff:ff
143.191.11.255  ether any         0      ff:ff:ff:ff:ff:ff
143.191.11.252  ether any         893    00:80:ad:02:62:6b
```

arp proxy <hostid> ether|token|<interface> <mac-addr>

The **arp proxy** command allows the NetBlazer to respond to ARP requests intended for another machine by supplying its own physical address. The NetBlazer then receives packets for that host and forwards them to the proper destination.

Using the **arp proxy** <hostid> <interface> command syntax is the preferred way to add a proxy ARP entry to the ARP list. This allows the NetBlazer to automatically get the physical address of the specified interface.

The **arp proxy** <hostid> ether|token <mac-addr> command syntax is provided for backward compatibility with earlier versions of the NetBlazer software.

Refer to the *NetBlazer Configuration Guide* for an example of how to use the proxy ARP function.

Asy Commands

The **asy** command takes you to the Configure Asy command level. To view a list of the commands available at this level, type a question mark (?) at the Configure Asy command level.

```
NB:Top> asy
NB:Top>Configure>Asy> ?
Available commands:
list            speed           top>            up>
?
```

Commands available at this level are used in conjunction with the Configure Line commands to monitor and control the NetBlazer serial lines.

asy list [<line-name>]

Entering an **asy list** command without the optional <line-name> parameter displays usage statistics for all serial lines.

```
NB:Top>Configure>Asy> list
Line       Speed   Bytes in  Bytes out  Overruns   Dropped     Queued Doing
line00      9600          0         59         0         0          0 connect
line01      9600          0         59         0         0          0 connect
line10     57600       1234        174         0         0          0 FREE
line11     19200       1501        177         0         0          0 FREE
line12      9600          0         59         0         0          0 connect
line13      9600          0         59         0         0          0 connect
line14      9600          0         59         0         0          0 connect
line15      9600          0         59         0         0          0 connect
line16      9600          0         59         0         0          0 connect
line17      9600        627       5817         0         0        163 Logged-In
```

To get additional information concerning a specific line, enter the **asy list** command with a <line-name> parameter.

```
NB:Top>Configure>Asy> list line17
line17        bytes interrupts     drops   overruns   bytes queued   fast bytes
input:          739        738         0          0              -            -
output:        8506       2637         0          -              0            0

9600 bps
dma queue: 0 bytes, 0 packets
dma fastqueue: 0 bytes, 0 packets
```

asy speed <line-name> <speed>

The **asy speed** command sets the data transfer rate on a specified serial port. Note that you also can use the **line speed** command to set the speed of the serial port.

The <speed> parameter may be set to any of the following standard data transfer rates: 300, 1200, 2400, 4800, 9600, 19200, 38400, 57600, or 115200.

Note: *The speed of the connected device must match the line speed.*

The following display shows you how to change the speed of line01 to 19.2K bps.

```
Top>Configure>Asy> speed line01 19200
```

If you are using a terminal connected to the specified line, the command prompt displayed after this command will be scrambled. Set your terminal to match the speed of the line (e.g., 19.2K bps), and press the return key. The prompt then is displayed correctly.

Bin Commands

The **bin** command takes you to the Bin command level, which comprises a collection of miscellaneous commands. To view a list of the commands available at this level, type a question mark (?) at the Bin command level.

```
NB:Top> bin
NB:Top>Bin> ?
Available commands:
commands        echo            edit            eval
grep            more            output          passwd
settime         sleep           source          tee
tcl             time            top>            tty
type            up>             uptime          version
wall            who             write           ?
```

bin commands [terse|verbose]

The **commands** command displays a complete list of the available NetBlazer commands. If you enter the optional terse parameter, all commands at a particular command level are displayed on a single line rather than on multiple lines. This difference is illustrated in the following example.

```
NB:Top>Bin> commands |grep Alias
Top>Configure>Alias>  add
Top>Configure>Alias>  delete
Top>Configure>Alias>  list
Top>Configure>Alias>  top
Top>Configure>Alias>  up
Top>Configure>Alias>  ?
NB:Top>Bin> commands terse |grep Alias
Top>Configure>Alias>  add delete list top up ?
```

bin echo [-n] <message>

The **echo** command prints a message to the user's screen. If you do not include the optional -n parameter, the message is followed by a new line. However, if you include the -n parameter, the new line is not sent. A couple of examples are shown below.

```
NB:Top> echo Hello NetBlazer users
Hello NetBlazer users
NB:Top> echo -n Hello NetBlazer users
Hello NetBlazer usersNB:Top>
```

Echo commands are useful in source files to give users feedback about what is happening when they run the **source** <filename> command.

Another use for the **echo** command is to set up a login message for a specific user by using **echo** in the user's login command, as illustrated below.

```
NB:Top> user command carol "echo Access denied - Call MIS department; logout"
```

bin edit [<filename>]

The **edit** command allows you to edit a local or remote file. Using the NetBlazer editor requires an understanding of the UNIX ed and ex line editor commands. Refer to the UNIX man pages for a description of these two commands.

Caution: *The **edit** command should only be used to edit ASCII files that contain 1000 lines or less.*

Do not attempt to edit TELEBIT.SYS, FRED, or any of the loadable module (.LDR or TOKEN002.MCD) files.*

Before editing any of the NetBlazer configuration files (STARTUP.CNF, MODEMCAP.TXT, CHAT.TXT or PASSWD), be sure to make a backup copy of the file.

To start editing a file, enter **edit** followed by the name of the file at the NetBlazer command prompt.

```
NB:Top>Bin> edit startup.cnf
"startup.cnf" 211 lines, 7851 characters
*
```

To edit a file on an remote IP host system, use **edit** <host>:<pathname> to specify the location and pathname of the file.

When the editor is started, it loads a copy of the file into a special area in the NetBlazer's memory called the buffer. Any changes made in the buffer do not appear in the original file until you enter a command to write (**w**) the contents of the buffer to disk.

To close a file without saving any changes, enter **q!** at the editor command prompt.

```
*q!
NB:Top>Bin>
```

The NetBlazer editor provides various line oriented editing commands that are used to display, change, delete or insert lines in the file's buffer. To view a summary of the editor commands, enter **h** at the editor prompt.

```
*h
(.)a              append after given line
(.,.)c            change lines
(.,.)d            delete lines
e[!] file                 edit new file
f file            file information or new file name
(1,$)g/pattern/cmd      execute cmd for all lines that match
h                 help information
(.)i              insert text
(.,.+1)j          join lines
(.)kx             create mark
(.)l              list lines
(.)p              print lines
q(!)              quit editting file
($)r [file]       read from file
(.,.)s /s1/s2/g substitute
set [no]number [no]ignorecase [no]prompt all
(1,$)v/pattern/cmd      execute cmd for all lines that don't match
version           display version number
(1,$)w[!] [file] write to file
(1,$)wq [file]  write to file and quit

line numbers (lnr)'s:
number            decimal line number
/pattern/         find pattern string
?pattern?         backwards find pattern string
$                 last line
.                 current line
lnr[+-][number] above + or - decimal number or 1
```

Users of the UNIX ed or ex line editors should recognize most of the commands listed above. Note, however, that not all of the features of ed or ex are implemented in the NetBlazer editor. For example, the use of regular expressions for pattern matching is not supported in the NetBlazer.

When using a line-oriented editor, you must specify which line or lines to operate on and the operation to be performed. If only one line number is specified, the operation is performed on that line. If two line numbers separated by a comma are specified, the operation is performed on the inclusive range of lines. For example, **1,5p** means print (display) the first five lines.

The **$** character is used to address the last line of a file. Thus, entering **1,$p** prints all the lines in a file.

The editor keeps track of the line on which you last performed some operation. This line is called the current line and is addressed by typing a dot character (a single period).

If you do not specify a line number, most edit operations are performed on the current line. For example, entering a **d** command without specifying a line number deletes the current line (the last line addressed).

If you do not specify an operation, the editor prints the line or lines specified by the line address. Therefore, entering **1,$** is equivalent to entering **1,$p**. If you press the return key without entering a line address or operation, the editor selects and displays the line following the current line.

You can specify a line number that is relative to the current line by using the **+** (plus) and **-** (minus) characters. For example, **+5** refers to the line five lines after the current line; **-3** refers to the line three lines before the current line. Entering a **+** or **-** without a specifying a number refers to the next or previous line.

One of the most convenient features of the editor is the ability to reference a line based on its content. The following display illustrates how to use this feature to find lines in the STARTUP.CNF file that contain the text pattern "route."

```
NB:Top>Configure> edit startup.cnf
"startup.cnf" 211 lines, 7851 characters
*/route/
#add static routes:
*//
configure ip route addprivate 127.0.0.1/32    loopback
*//
configure ip route addprivate 143.191.20.1/32    tbupdate
*//
configure ip route add 143.191.11.0/24    en0
*//
#add static routes:
```

The editor starts at the line following the current line and searches forward in the buffer looking for the first occurrence of the text pattern "route." If the end of the buffer is reached, the editor wraps around to the beginning, and continues searching until the pattern is found or it ends up back at the current line. If the text pattern is not found, the editor displays the message Pattern not found.

Notice that entering **//** is a shorthand way to search for the next occurrence of a previously specified text pattern.

You can search backward through the buffer by enclosing the search string between question marks in place of slashes, as shown below.

```
*?route?
configure ip route add 143.191.11.0/24    en0
```

The next several examples shows you how to create and edit a new file. The new filename must conform to DOS file naming conventions – eight alphanumeric characters with a three character extension. Do not use control characters or special symbol such as: *?:;,=+<>|\"[]\t

```
NB:Top> edit example.src
Can't open file 'example.src'
*
```

Because this is a new file that does not exist on the disk, the editor complains that it cannot open the file. If you get this message when you attempt to edit an existing file, it indicates that you probably have made a mistake when entering the filename.

The asterisk (*) following the opening message is the editor's command prompt, which tells you that the editor is prepared to accept commands.

When editing a new file, the first thing to do is to start adding text to the file using the append (**a**) command. Whenever you use a command that requires you to enter new text, the editor goes into input mode and waits for you to type the new text.

```
*a
echo This is the first line
echo This is the second line
echo This is the last line
.
*
```

To let the editor know that you are finished entering text, you must type a period at the beginning of a line, followed immediately by a carriage return. This returns you to the command prompt, where you can enter commands again.

Next, let's assume that you want to add a new line before the first line in the file. To do this you have two choices. You can use the **0a** command to add the new line *after* line 0, or you can enter **1i** to insert the new line *before* line 1. The next display shows you how to insert a new line before line 1.

```
*1i
echo This is the new first line
.
*
```

As shown in the following display, you must now change the second and third line in the buffer to read correctly.

```
*1,$
echo This is the new first line
echo This is the first line
echo This is the second line
echo This is the last line
*
```

One way to do this is to use the change (**c**) command to replace the second and third line.

```
*2,3c
echo This is the new second line
echo This is the new third line
.
*
```

Another way to do the same thing is to use the substitute (**s**) command to edit the second and third line, as shown below.

```
*2s/first/new second/
*3s/second/new third/
```

These two commands tell the editor to replace "first" with "second" in line 2 and replace "second" with "third" in line 3.

Displaying the buffer contents shows that the file is now correct and ready to be saved to disk.

```
*1,$
echo This is the new first line
echo This is the new second line
echo This is the new third line
echo This is the last line
*
```

The write (**w**) command saves the current contents of the buffer to disk and returns the command prompt to allow you to continue using the editor.

```
*w
"example.src" 4 lines, 116 characters
*
```

If you are finished editing, use the **wq** command to save the file to disk and quit out of the editor.

```
*wq
"example.src" 4 lines, 116 characters
NB:Top>
```

bin eval <command>

The **eval** command is used in **tcl** scripts to evaluate **netblazer** commands that include pipes or aliases. See the **tcl** command description later in this section for a description of **tcl** scripts.

bin grep <string> [<filename>]

The **grep** command searches a file or the piped output of a command for all occurrences of a specified string.

To display all lines in a file that contain a specified string, use the **grep** <string> <filename> command. For example, the following command displays a list of all the **route** commands in the STARTUP.CNF file.

```
NB:Top> grep route startup.cnf
#add static routes:
configure ip route addprivate 143.191.20.1/32    tbupdate
configure ip route addprivate 127.0.0.1/32    loopback
configure ip route add 143.191.10.0/24    en0
configure ip route addprivate 143.191.22.0/24    en0  143.191.10.29  3
configure ip route add default en0  143.191.10.253  4
```

To scan the output of a NetBlazer command for lines containing a specified string, use the **|grep** <string> command. The next example displays a list of routes that use IP address 143.191.10.29 as a gateway.

```
NB:Top> route list |grep " 143.191.10.29"
143.191.13.0    24  en0          143.191.10.29    2        P 0       0
143.191.22.0    24  en0          143.191.10.29    3        P 0       0
```

When searching for a <string> that includes one or more spaces, you must enclose the string in quotes, as illustrated in the above screen. When searching for a <string> that includes an exclamation point, you must enter it as **\!** to avoid confusion with a **history** reference.

If the output of **grep** produces more than a screenful of data, you can pipe it to **more** to display one screenful at a time (e.g., **route list |grep 11.2 |more**).

Not all commands can be piped through **grep**; in particular, any command that starts a session (e.g., **ping** or **telnet**) and **trace** commands cannot be piped through **grep**.

bin more [<filename>]

The **more** command displays the content of a file or the piped output of a command one screenful (23 lines) at a time. When the screen is full, you are prompted to press any key except the **q** key to display the next screenful. Pressing the **q** key stops the display and returns the command prompt.

To display the contents of a file, use the **more** <filename> command. For example, entering **more passwd** displays the contents of the PASSWD file one screenful at a time.

To display the output of a NetBlazer command one screenful at a time, use the **|more** command. For example, entering **route list | more** pipes the output of the **route list** command through **more**.

Not all commands can be piped through **more**; in particular, any command that starts a session (e.g., **ping** or **telnet**) and **trace** commands cannot be piped through **more**.

bin output [-a] <filename>

The **output** command sends the piped output of a command to a file that you specify in the command string. You can specify a new or existing file name. If you select a new file name, the **output** command creates that file and stores the piped output there. If you specify an existing file, the new information writes over the existing file unless you precede the file name with the -a parameter. If you include -a in your command string, the **output** command appends the piped output to the end of the file.

The following example shows how to use the **output** command to save the current interface trace settings.

```
NB:Top>Configure>Interface> tracedump |output trace.src
```

Later, you can restore the current interface trace settings by entering a **source trace.src** command.

bin passwd [<username>]

The **passwd** command allows all local users to change their password, provided the system administrator has enabled this feature by issuing a **user changeable on** command.

Note: *This command cannot be used to change the passwords of remote users (i.e. those users whose user records are kept in a passwd file on a remote host). Remote user records may only be changed on the remote system.*

Normal (nonprivileged) users can change their password by logging into the NetBlazer and entering the **passwd** command. They are then prompted for their old password and the new password.

```
NB login: carol
Password:
NB:Top> passwd
Changing password for carol
Enter old password :
New Password :
Retype Password :
```

The optional <username> parameter may only be entered by a user with sysconf privileges to change the password of another user.

```
NB login: root
Password:
NB:Top> passwd carol
Changing password for carol
New Password :
Retype Password :
```

This has the same result as using the **user password** command to change a user's password.

bin settime

The **settime** command sets the date and time used by the NetBlazer.

```
NB:Top>Bin> settime
Current time is Sun Oct 11 19:25:13 1992
[you may enter a non-numeric value (e.g. "cancel") to cancel data entry]
Enter the year [92]:
Enter the month number (1 to 12) [10]:
Enter the day of the month (1 to 31) [11]:
Enter the hour (0 to 23) [19]:
Enter the minute (0 to 59) [25]:
Enter the current second (0 to 59) [13]:
```

Entering a **time** command displays the current date and time.

bin sleep <delay>

The **sleep** command tells the NetBlazer to wait the number of seconds specified by the <delay> parameter before processing the next command. This command is useful in **source** command files when a delay in the command processing is required to allow the NetBlazer to complete a process before continuing.

bin source [-v] <command-file>

The **source** command automatically runs a series of commands contained in the file specified by the <command-file> parameter. The -v (verbose) option displays the source lines as they are executed for debugging purposes.

Non-privileged users can only source a file that ends in .usr (e.g., **source foo.usr**). There are no restrictions, however, on **source** commands issued from a user's login command.

To use this command, you must first create a command file using a text editor. Three ways to do this are:

1. Create a command file on the NetBlazer system diskette using the **edit** command.
2. Create the command file using the text editor of an IP host system that can be reached by the NetBlazer through the network; then either use FTP to copy the file to the NetBlazer diskette or run the command file directly from remote system.

To run a **source** command file located on a remote IP host, use **source** <host>:<filename> where <host> is the host ID or address of the remote system and <filename> is the full pathname of the file.

3. Create the command file on an MS-DOS system that supports 3.5 inch high-density diskettes, then use the MS-DOS copy utility to copy the file onto the NetBlazer diskette.

Commands in the command file must be set up to run from the Top command level and include all required parameters.

Do not use commands, like **ipdial**, that require operator input. Also, avoid using commands, such as **reboot** or **shutdown**, which may have adverse results.

The order in which the commands are entered is important. Because the commands are executed in sequence starting at the top and working down through the file, commands that define parameters used by other commands should be entered earlier in the list.

bin tcl

The **tcl** command allows system administrators, who are experienced with the UNIX Tcl command language, to set up complex command scripts.

Issuing a **tcl** command at the command prompt places the NetBlazer in interactive **tcl** command mode. A percent sign (%) is displayed in front of the command prompt to indicate that the NetBlazer is ready to accept **tcl** commands. To return the NetBlazer to normal command mode, enter a **system** or **netblazer** command.

```
NB:Top> tcl
% NB:Top> system
NB:Top>
```

To issue **tcl** commands from a script file, put a **tcl** command at the top of the file, as shown in the following example.

```
tcl
# get time of day from time command
set time [split [system time] " "]
# get hour of day
set hour [lindex [split [lindex $time 7] ":"] 0]
# get tty line number
set tty [system tty]
if {$tty=="line00" || $tty == "line01" } {
     if {$hour >= 9 && $hour < 17} { echo restricted time for $tty; logout }
     }
```

This **tcl** script restricts dial-in access to the NetBlazer between the hours of 9:00 AM and 5:00 PM.

Caution: *If you are having trouble understanding the logic in the above example, read the UNIX man pages on Tcl for information on how to use the Tcl scripting language.*

The NetBlazer supports the following standard UNIX Tcl commands: append, array, break, case, catch, concat, continue, error, eval, expr, for, foreach, format, global, history, if, incr, info, join, lappend, lindex, linsert, list, llength, lrange, lreplace, lsearch, lsort, proc, regexp, regsub, rename, return, scan, set, split, string, trace, unset, uplevel, upvar, while.

The following NetBlazer specific **tcl** commands are also provided.

echo <arguments> – sends the arguments to the terminal.

netblazer [-s] <command> <arguments> – invokes a NetBlazer command with the given arguments. Any output from the command goes to the terminal in the normal way. If -s is specified, then the return code from the command is returned to **tcl**. 0 indicates that the command was executed successfully, > 0 indicates that an error has occurred, and -1 indicates a bad usage.

system <command> <arguments> – invokes a NetBlazer command with the given arguments. Any output from the command is returned as a **tcl** result string.

ttyask <prompt> – displays the prompt and returns the resulting line typed by the user.

unknown – Any unknown **tcl** command is passed to the normal NetBlazer command processor as a **netblazer** command.

bin tee [-a] <filename>

The **tee** command sends the piped output of a command to a file that you specify in the command string and displays it on your terminal screen. You can specify a new or existing file name. If you select a new file name, the **tee** command creates that file and stores the piped output there. If you specify an existing file, the new information writes over the existing file unless you precede the file name with the -a parameter. If you include -a in your command string, the **tee** command appends the piped output to the end of the existing text in the file.

The following example shows how to use the **tee** command to save the current interface trace settings.

```
NB:Top>Configure>Interface> tracedump |tee trace.src
trace en1 none; trace en1 disabled input output headers ip off
```

Later, you can restore the current interface trace settings by entering a **source trace.src** command.

bin time

The **time** command displays the current date and time. To change the date and time, enter a **settime** command. You will be prompted for the new date and time.

bin tty

The **tty** command displays the line or port that you are currently logged in through.

```
NB:Top> tty
143.191.12.26:10935
```

To get a list of users that are currently logged into the NetBlazer, use the **who** command.

bin type <filename>

The **type** command sends the entire specified text file to your terminal screen. You can view, but not alter, the text. This command is similar to the **more** command, which sends one screenful of text at a time.

bin uptime

The **uptime** command displays the current date and time, the number of hours and minutes the NetBlazer has been running since the last time it was started, and the number of users currently logged onto the system.

bin version

The **version** command displays information about the current version of the NetBlazer software.

bin wall [<message>]

The **wall** command immediately sends a message to all users currently logged in to the NetBlazer. You can enter a single line message or a multi-line message consisting of up to 990 characters.

The following example shows you how to enter a single line message.

```
NB:Top> wall "Hello NetBlazer users."

Message from root on 192.124.95.18:11329
Hello NetBlazer users.
```

When entering a single line message that contains an exclamation point, you must enter it as \! to avoid confusion with a history reference.

The next example shows you how to enter a multiline message.

```
NB:Top> wall
Enter the message for 'wall' (up to 990 characters).
Type . at the start of a line to end the message.
Line 1: The NetBlazer will be shut down 30 minutes from now.
Line 2: Please finish what you are doing and log out.
Line 3: .

Message from root on 192.124.95.18:11329
The NetBlazer will be shut down 30 minutes from now.
Please finish what you are doing and log out.
```

Note that the **shutdown** command also sends a warning message to notify all users that the NetBlazer is being shut down.

To send a messages to specific users, use the **write** command.

bin who [am i]

The **who** command displays a list of users who are currently logged into the NetBlazer.

```
NB:Top> who
user       time           from
root       Dec 16 14:41   line00
root       Dec 17 14:03   143.191.12.26:10935
```

If the same account is used multiple times as shown above, and you want to know which line or port you are logged in on, you can enter a **who am i** command to get just your login information.

```
NB:Top> who am i
user      time           from
root      Dec 17 14:03   143.191.12.26:10935
```

bin write <user> [<location>]

The **write** command immediately sends a message to all users logged in with the user ID specified by the <user> parameter, unless the <location> is specified. In this case, only the user logged in from the specified location is sent the message.

For example, to send a message to all users logged in as root, issue a **write root** command as illustrated below.

```
NB login:  root

NB:Top> who
user      time           from
root      Dec 16 14:41   line00
root      Dec 18 05:50   143.191.12.26:12104
NB:Top> write root
Enter the message for 'write' (up to 990 characters).
Type . at the start of a line to end the message.
Line 1: Root users please login using your own IDs
Line 2: so that we know who you are.
Line 3: .

Message from root on 143.191.12.26:12104
Root users please login using your own IDs
so that we know who you are.
```

The next example shows you how to send a message to just the user who is logged in from a specific location.

```
NB:Top> who am i
user      time           from
root      Dec 18 05:50   143.191.12.26:12104
NB:Top> write root 143.191.12.26:12104
Enter the message for 'write' (up to 1024 characters).
Type . at the start of a line to end the message.
Line 1: This message is to myself.
Line 2: .

Message from root on 143.191.12.26:12104
This message is to myself.
```

To send a message to all logged-in users regardless of their user IDs, use the **wall** [<message>] command.

Boot Command

The **boot** [list] | [normal] | [bootp [<delay>]] command sets or shows the method used to boot the NetBlazer. You can configure the NetBlazer to do a normal boot from the FRED file on the system diskette or boot from a UNIX host using BOOTP.

Note: *When you change the setting of the **boot** command, be sure to issue a **save** command before rebooting the system. This writes the current boot setting to the NetBlazer's nonvolatile memory.*

boot [list]

Entering a **boot** command by itself or with the **list** option displays the current boot setting.

```
NB:Top>Configure> boot list
boot normal
```

Initially, the NetBlazer is set to do a normal boot using the FRED file on the system diskette.

boot bootp [<delay>]

Entering a **boot bootp** command configures the NetBlazer to boot the BOOTP file instead of the standard system.

```
NB:Top>Configure> boot bootp 60
boot bootp 60 (seconds)
```

This allows the to NetBlazer use BOOTP and TFTP client services to boot from a UNIX host that is connected to the same Ethernet segment as the NetBlazer's en0 interface.

The optional <delay> parameter indicates how long the NetBlazer should attempt to boot over the network before reverting to booting FRED from the diskette. If no delay is specified, the NetBlazer uses the default value of 300 seconds.

For instruction on how to boot the NetBlazer from a UNIX host on the network, refer to the chapter on Network Management in the *NetBlazer Configuration Guide*.

boot normal

Entering a **boot normal** command configures the NetBlazer to do a normal boot from the FRED file on the NetBlazer system.

Configure Command

The **configure** command takes you to the Configure command level. To view a list of the commands available at this level, type a question mark (?) at the Configure command prompt.

```
NB:Top> configure
NB:Top>Configure> ?
Available commands:
alias>          appletalk>      arp>            asy>
boot            debug>          destination>    dialout>
files           group>          hostname        interface>
ip>             ipx>            line>           message>
msglog          phonelist>      ppp>            save
security>       servers>        setconsole      slip>
snmp>           syslog>         top>            up>
user>           ?
```

For a description of the above configuration commands, refer to the specific command heading. For example, to read about the **dialout** commands, look under Dialout Commands in this manual.

All commands at or beneath the Configure command level are available only to users who have been given sysconf or sysstat privileges. User with sysconf privileges can use any of the commands at or below this level to configure the NetBlazer. Users with sysstat privileges are restricted to displaying information about the current configuration of the NetBlazer; they cannot change the configuration.

Caution: *The **debug** commands are provided primarily to aid Telebit in isolating problems in the NetBlazer. Do not use the **debug** commands without first contacting Telebit.*

Debug Commands

The **debug** command takes you to the Configure Debug command level. To view a list of the commands available at this level, type a question mark (?) at the Configure Debug command level.

```
NB:Top> debug
NB:Top>Configure>Debug> ?
Available commands:
alloc           args            autoconfigure   clear
core            db              devs            dh
display         dl              dump            files
filter>         gdt             hardware        hex
hogmemory       ident           idle            idt
inb             irq             kill            maplist
memory          memset          modemcap        noautoconfigure
nocore          notnclient      notnserver      outb
printf          process         ps              psignal
remote          sessions        show            status
time            timers          tnclient        tnserver
top>            unmap           up>             watch
writebyte       writelong       writeshort      ?
```

Caution: *These commands, provided to help Telebit resolve software problems, are unsupported and likely to change without notice. Because many of these commands may have undesirable effects, you should not use commands at this level unless instructed to do so by Telebit's Technical Support department.*

debug alloc [<value>]

The **debug alloc** command sets the allocation debug level, which is only available if memory allocation debugging is compiled into the NetBlazer software. The <value> parameter is a summation of any of the following options:

1	Set and check a fence value on malloc and free
2	Set memory after free
4	Set memory after alloc
8	Set watch point on fence after data
16	Set watch point on fence

Note that settings 2 and 4 degrade performance.

If the <value> parameter is not entered, the current option settings are displayed.

debug args

The **debug args** command displays the argument list.

debug autoconfigure

The **debug autoconfigure** command sends debug messages to the console from the line autoconfigure function. The **debug noautoconfigure** command stops sending line autoconfigure messages.

debug clear

The **debug clear** command clears any watch points.

debug core

The **debug core** command turns on core dumps. The **debug nocore** command turns off core dumps.

debug db <address> [<count>]

The **debug db** command is the same as the **display** command, except that the memory is displayed as 8-bit hex quantities.

debug devs [verbose]

The **debug devs** command lists logical names of devices that the NetBlazer software supports. The verbose option provides additional information about the logical devices.

debug dh <address> [<count>]

The **debug dh** command is the same as the **display** command, except that the memory is displayed as 16-bit hex quantities.

debug display <address> [<count>]

The **debug display** command displays the specified memory address in 32-bit hex words and ASCII for the <count> values. Addresses that are invalid are rejected.

debug dl <address> [<count>]

The **debug dl** command is the same as the **display** command.

debug dump

The **debug dump** command forces a core dump and then reboots the system. This command records the current state of the NetBlazer for later analysis by Telebit Technical Support. Entering a control-alt-f10 on an AT keyboard also forces a dump.

debug files

The **debug files** command displays a list of open files.

debug filter

The **debug filter** command takes you to the Configure Debug Filter command level. These commands, which work the same as the standard packet filtering commands, filter the **interface trace** output for one or more interfaces. Refer to the **interface trace** and **filter** command descriptions for additional information.

debug gdt <start> [<end>]

The **debug gdt** command displays the global descriptor table entries ranging from <start> to <end>.

debug hardware

The **debug hardware** command provides information about the NetBlazer's hardware configuration.

debug hex <filename>

The **debug hex** command produces a hex dump listing of the specified file.

debug hogmemory <bytes>

The **debug hogmemory** command grabs the specified number of bytes from memory.

Caution: *This command only should be used in a test environment to force memory shortages for testing memory stress situations.*

debug ident <filename> ...

The **debug ident** command displays the revision level of the specified files.

debug idle

The **debug idle** command displays the CPU idle percentage over the last second and the number of times timerproc was skipped.

debug idt <start> [<end>]

The **debug idt** command displays the interrupt descriptor table entries that range from <start> to <end>.

debug inb <port> <count>

The **debug inb** command displays the contents of an I/O port or ports if count is specified.

Caution: *This command can be dangerous, as some ports do unpredictable things if they are read. This can bring down the system or confuse the drivers.*

debug irq <value>

The **debug irq** command displays the status of the I/O interrupt that is specified by a value between 0 and 15.

debug kill <pid>

The **debug kill** command kills most processes. You are not allowed to kill system processes (e.g., main, timer, network, tracer, or killer).

Caution: *Do not use this command unless Telebit's Technical Support department instructs you to do so.*

debug maplist <addr> <count>

The **debug maplist** command displays the memory management map for the given address and page count.

debug memory <command>

The **debug memory** command invokes memory commands that are now available as **debug** commands. For example, **debug alloc** and **memory alloc** are equivalent commands.

debug memset <addr> <char> <length>

The **debug memset** command sets the memory at <address> to the hex byte value specified by <char> for the number of bytes specified by the <length> parameter.

Caution: *Do not use this command unless Telebit's Technical Support department instructs you to do so.*

debug modemcap [<modem>]

The **debug modemcap** command displays the MODEMCAP.TXT entry for a specified modem. If you do not enter the optional <modem> parameter, the NetBlazer displays a list of modems supported by the MODEMCAP.TXT file.

debug noautoconfigure

The **debug noautoconfigure** command stops sending line autoconfigure messages to the console.

debug nocore

The **debug nocore** command turns off core dumps.

debug notnclient

The **notnclient** command turns off Telnet client option negotiation debugging.

debug notnserver

The **notnserver** command turns off Telnet server option negotiation debugging.

debug outb <port> <value>

The **debug outb** command sends <value> to the I/O port specified by the <port> parameter.

debug printf <fmt> <value ...>

The **debug printf** command displays <value ...> in the specified format using the tprintf routine.

debug process [<pid>]

The **debug process** command displays the information in the proc structure for the process specified by <pid>, or for the current process if you do not enter <pid>.

debug ps

The **debug ps** command invokes the **list ps** command. This command is here so that the command interpreter does not interpret **ps** as an abbreviated **psignal** command.

debug psignal <value> <count>

The **debug psignal** command wakes up the <count> processes waiting on <value>.

Caution: *Do not use this command unless Telebit's Technical Support department instructs you to do so.*

debug remote [on|off] [en0|en1|en2] [file <filename>] [host <hostid>] [ether <hardware-addr>] [myip <myip>] [list] [test]

The **debug remote** command configures the NetBlazer to send core dumps to a remote IP host through a specified Ethernet interface.

If you just turn remote dumping on and do not supply the host's hardware address and filename, the BOOTP protocol gets the required information from a BOOTP server. If there is no BOOTP server or the BOOTP server is down, then the network dump fails, and the core dump is sent to a CORE.DMP file on the NetBlazer diskette.

The actual dump is sent to a remote host by TFTP, which means that there must be a TFTP server running on the remote host. Note that TFTP requires an existing file that can be written to by everybody.

Before using the remote dump facility, verify that TFTP is functioning on the remote host.

If you wish to take dumps over the network without using BOOTP, you must enter the necessary information such as the ip address, the hardware address, and the filename with the **debug remote** commands.

The **debug remote on|off** command turns the remote core dump function on or off.

The **debug remote en0|en1|en2** command specifies which Ethernet interface to use when sending a core dump to a remote host. If you do not specify an interface, en0 is used by default.

The **debug remote host** <ip> command specifies the IP address or domain name of the TFTP server.

The **debug ether** <hardware-addr> command specifies the Ethernet address of the TFTP server or the gateway to reach the TFTP server.

The **debug remote myip** <myip> command specifies the IP address of this NetBlazer. Normally, this command is not required because the NetBlazer's global IP address is used.

The **debug remote list** command displays the current remote core dump information.

The **debug remote test** command tests the remote core dump function. The NetBlazer is disabled until the test finishes.

debug sessions

The **debug sessions** command displays the number of foreground and background sessions currently running on the NetBlazer.

debug show

The **debug show** command shows the internal console screen log, which keeps the last 4096 bytes that have been written to that screen. This is particularly useful from remote locations and when there is no external console terminal.

debug status [arguments]

The **debug status** command runs a status monitoring session on a terminal or on the internal console screen. The status display currently must be run on a terminal that supports the ANSI escape sequences.

Issuing a **debug status start** command starts a standard status monitoring session, which displays the status of all items available on this NetBlazer, as illustrated in the following example.

```
host      pearl          1.91x1-beta    time      11:04:03   date      11/19/92
          *-------------------------------------------------*  Value  %
Units
cpu       |X                                                |    6    3%
ints      |0000                                             |  205
line00    |                                                 |    0    0% C/S
line01    |                                                 |    0    0% C/S
syn0      |                                                 |    0    0% C/S
en0       |                                                 |    5    1% P/S
ibuf      |LLLLLLLLSSSSSSSSSSSSSSS                          |  239   47% #
mbuf      |                                                 |    9    1% #
memfree   |XXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                   |  765K  60%
fd        |                                                 |    0    0%
          *-------------------------------------------------*
```

The maximum values of various status fields (ints, ethernet, token ring) are nominal values that are chosen to be reasonable in most cases.

By default, the system updates this display each second and refreshes (redraws) it after every tenth update.

To stop the status monitoring session, escape out of the session (e.g., enter ctrl-] <return>), then issue a **session close** or **debug status clear** command to close the session and reset the status monitor.

The **delay** <value> and **refresh** <value> arguments control how often the system updates and redraws the status display. The delay value specifies the delay in seconds between updates, and the refresh value specifies the number of updates to display before refreshing (redrawing) the display. For example, a **debug status start delay 2 refresh 5** command starts a standard status monitoring session that is updated every two seconds and refreshed after every fifth update.

The **internal** option redirects the status monitor output to a previously installed internal video board and monitor.

The **debug status list** command displays the status monitor's current settings.

You can build a custom status display using the following command syntax:

```
debug status <item> [line <line>] [col <column>] [notitle] [maximum <value>]
```

Items that can be shown in the status display are defined in Table DEB-1.

Table DEB-1. Status Display Items

Item	Description
host	Show the host name of the NetBlazer.
time	Show the current time.
date	Show the current date.
text <string>	Display the text string. This string must be enclosed in quotes.
cpu	Show the CPU time used.
ints	Show the interrupts status.
memfree	Show the current available memory.
ibuf	Show the interrupt buffers.
mbuf	Show the mbufs.
fd	Show the floppy disk I/O.
asyXX asy <line-name>	Show the transmit and recieve status of a specified line. To display all of the asynchronous lines, use the asys command.
iface <interface> interface <interface>	Show the packets transmitted for a specified interface. To display all of the interfaces, use the nets command.
asys	Show the status for all of the NetBlazer's asynchronous lines.
nets	Show the status for all of the NetBlazer's network adapters.

Each item is displayed in the order entered. The **line** <line> and **col** <column> parameters set the position of an item.

The **notitle** option suppresses the title that is normally displayed.

The **maximum** <value> parameter specifies a value to override the item's default value.

The following screen shows the **debug status** commands used to build the standard screen.

```
NB:Top>Configure>Debug> status standard
config debug status host line 1
config debug status time line 1 col 40
config debug status date line 1 col 60
config debug status text "*-------------------------------------------------
*"
col 10
config debug status text "Value  % Units" col 64 line 2
config debug status cpu
config debug status ints
config debug status asys
configure debug status line00
configure debug status line01
config debug status nets
configure debug status interface en0
configure debug status interface en1
config debug status ibuf
config debug status mbuf
config debug status mem
config debug status fd
config debug status text "*-------------------------------------------------
*"
col 10
```

debug time <command>

The **debug time** command displays how long it took to execute a specified command.

debug timers

The **debug timers** command displays the current pending timers.

debug tnclient

The **debug tnclient** command turns on Telnet client option negotiation debugging.

debug tnserver

The **debug tnserver** command turns on Telnet server option negotiation debugging.

debug unmap <address> <count>

The **debug unmap** command unmaps the specified address from the address space. About the only address that can usefully be unmapped is page zero, which is already unmapped.

Caution: *Do not use this command unless Telebit's Technical Support department instructs you to do so.*

debug watch <addr> <type> [124] [WRE]

The **debug watch** command sets a watch point at the address of the given type. The type should consist of [erw][124] (access type, length).

Caution: *Do not use this command unless Telebit's Technical Support department instructs you to do so.*

debug writebyte <address> <value>

The **debug writebyte** command writes a byte value into the specified address.

Caution: *Do not use this command unless Telebit's Technical Support department instructs you to do so.*

debug writelong <address> <value>

The **debug writelong** command writes a long value into the specified address.

Caution: *Do not use this command unless Telebit's Technical Support department instructs you to do so.*

debug writeshort <address> <value>

The **debug writeshort** command writes a short value into the specified address.

Caution: *Do not use this command unless Telebit's Technical Support department instructs you to do so.*

Destination Commands

The **destination** command takes you to the Configure Destination command level. To view a list of the commands available at this level, type a question mark (?) at the Configure Destination command level.

```
NB:Top> destination
NB:Top>Configure>Destination> ?
Available commands:
add             delete          gid             host
list            longname        top>            type
up>             ?
```

This set of commands allows you to create and maintain destination groups for the users. Assigning the users to a destination group has the following advantages:

- Simplifies the user's interaction with the NetBlazer by allowing a user to select a telnet/rlogin or dialout destination from a list of destinations assigned to that user.
- Allows you to restrict the user's access to a limited group of telnet/rlogin hosts and/or dialout destinations.

The **user destgroup** command assigns a user to a defined destination group. To start a dialout, telnet, or rlogin session with a destination contained in the user's assigned destination group, he or she issues a **session connect** command.

destination add [<dest-group>] [<short-name>]

The **destination add** command creates a new destination group or adds a new destination name to an existing destination group.

The <dest-group> parameter is the name of an existing or new destination group. If the <short-name> variable is entered, the new destination name specified by <short-name> is added to the designated destination group.

The <short-name> is used in the **session connect** <destination-name> command to start a dialout, telnet, or rlogin session with the specified destination.

If you enter the **destination add** command without any parameters, the NetBlazer begins a dialog. This dialog collects the information normally acquired by the **destination add**, **gid**, **host**, **longname**, and **type** commands. This makes configuration easier and helps prevent mistakes caused by forgetting to enter a necessary command.

destination delete <dest-group> [<dest-number>]

The **destination delete** command deletes an entire destination group when only the <dest-group> parameter is entered. If both parameters are entered, this command removes a destination entry from the specified group. The <dest-group> variable identifies the group to modify, and the <dest-number> parameter identifies the particular entry in the group to delete.

destination gid <dest-group> <group-id>

The **destination gid** command sets or changes the group ID number assigned to the specified destination group. The <dest-group> variable identifies the group to modify and the <group-id> parameter is the GID number assigned to the group.

destination host <dest-group> <dest-number> <dialout> | <telnet-host> | <rlogin-host>

The **destination host** command adds or changes the dialout, telnet, or rlogin destination information of a destination group entry.

The first two parameters, <dest-group> and <dest-number>, identify which entry within a specified group to assign a dialout destination or telnet/rlogin host ID.

The last parameter, <dialout> | <telnet-host> | <rlogin-host>, identifies the dialout destination, telnet host ID, or rlogin host ID, depending on whether the destination entry is a dial, telnet, or rlogin type destination. The telnet or rlogin host ID may be either a domain name or IP address.

destination list [<dest-group>]

The **destination list** command displays a list of the available destination groups or a list of the destinations in a particular group, depending on if the <dest-group> parameter is entered.

destination longname <dest-group> <dest-number> <long-name-string>

The **destination longname** command assigns a more descriptive name to a particular destination entry.

The first two parameters, <dest-group> and <dest-number>, identify which entry within a specified group to assign a new description.

The <long-name-string> parameter is the descriptive text to associate with the designated destination entry.

destination type <dest-group> <dest-number> dial | telnet | rlogin

The **destination type** command specifies the type of destination (i.e., dial, telnet, or rlogin) associated with a particular destination group entry.

The first two parameters, <dest-group> and <dest-number>, identify which entry within a specified group to modify.

The last parameter must be dial, telnet, or rlogin. Note that the host assigned to the designated entry must be appropriate for the type selected.

Dialout Commands

The **dialout** command takes you to the Configure Dialout command level. To view a list of the commands available at this level, type a question mark (?) at the Configure Dialout command level.

```
NB:Top> dialout
NB:Top>Configure>Dialout> ?
Available commands:
add             characteristics chat            crypto
delete          ditto           list            maxwait
minwait         password        phone           timeout
top>            uname           up>             ?
```

Commands at this level maintain a list of dial-out destinations that are used to establish a connection with a remote site in either packet or character mode.

Each dial-out destination can include multiple dial-out entries. If a given dial-out destination name includes more than one entry (phone number), the NetBlazer chooses the least recently used phone number to place a call. This prevents the NetBlazer from always trying to dial the first number when it already has been used and is busy.

dialout add [<dialout>] [<phone-number>] [<characteristics-group>] [<timeout>] [<max>] [<min>]

The **dialout add** command creates a new entry in the dial-out list.

The <dialout> parameter is a single word name that identifies the dial-out destination. If the dial-out destination is another NetBlazer with more than one modem connected, you should enter the same destination name for each telephone line connected to the remote NetBlazer.

The <phone-number> parameter is the telephone number of the remote destination.

The <characteristics-group> parameter specifies the characteristics group assigned to the dial-out destination record. The chosen characteristics group must be previously defined via the **group** commands.

The <timeout> parameter determines the number of minutes that an inactive character mode connection will be maintained. When a line is inactive for the number of minutes specified by the <timeout> value, the NetBlazer drops the telephone connection. If you set the <timeout> to zero, the connection will never time out.

The <max> parameter sets the maximum number of seconds that a dial-out record waits between calls. Setting this parameter to its default value of 0 disables the wait time function.

The <min> parameter sets the minimum number of seconds that a dial-out record waits between calls.

If this command is entered without any parameters, the NetBlazer prompts you for the necessary information as illustrated below.

```
NB:Top>Configure>Dialout> add
Name of dialout: branch
Phone number: 555-1234
Line characteristics [dialout]:
Char mode timeout (in minutes) [60]:
Dialout: name=branch, phone=555-1234, characteristics=dialout
Okay (yes|no|quit) [y]?
```

dialout characteristics <dialout> [#<item-number>] [<group>]

The **dialout characteristics** command sets or changes the characteristics group assigned to a dial-out destination.

The <dialout> and optional #<item-number> parameters identify the dial-out destination to which to assign the designated characteristics group.

The <group> parameter identifies the characteristics group assigned to the selected dial-out destination. The chosen group name must be previously defined via the **group** commands. If no <group> is entered, the characteristics group for this dial-out destination is set to <none>.

The NetBlazer comes with seven standard line characteristic groups: dialout, v32dialout, pepdialout, request, raw, raw_dial, and v25dialout. Refer to the **line characteristics** command description for a description of these characteristic groups.

dialout chat <dialout> <chat-script-name>

The **dialout chat** command sets or changes the chat script assigned to a dial-out destination.

The <chat-script-name> parameter specifies which chat script in the CHAT.TXT file to use when attempting to establish a dial-out connection with the remote site. The **ics** chat script is the default chat script used to dial out to another NetBlazer at the remote site.

dialout crypto <dialout> [<crypto-key>]

The **dialout crypto** command sets or changes the crypto key assigned to a dial-out destination. The <crypto-key > parameter is the crypto key, which is used to encrypt a crypto response when a crypto challenge is received from the remote site. The remote system must have a user record that matches the uname and crypto key assigned to this dial-out record.

This command changes the crypto key for all dial-out records with the same name as the <dialout> parameter. If no <crypto-key> is entered, the crypto key is removed from the dial-out record(s).

dialout delete <dialout> [#<item-number>]

The **dialout delete** command deletes the dial-out destination specified by the <dialout> and #<item-number> variables. The #<item-number> parameter is only required when more than one entry share the same dial-out destination name.

Note: *Deleting a dial-out record with the same name as a dynamic interface disables dial-out connections through that interface.*

Deleting a dynamic interface record removes dial-out and user records with the same name as the interface.

dialout ditto <dialout> [#<item-number>] <phone-number>

The **dialout ditto** command copies a dial-out destination record and assigns a new telephone number to the duplicate record. This provides a convenient way to create the dial-out records for a destination site with multiple phone numbers.

The <dialout> and optional #<item-number> parameters select the dial-out record to copy.

The <phone-number> parameter is the telephone number assigned to the duplicate record.

dialout list [<dialout>] [#<item-number>]

The **dialout list** command displays either a list of the currently defined dial-out destinations, or the settings of the dial-out destination specified by the optional <dialout> and #<item-number> parameters.

dialout maxwait <dialout> <seconds>

The **dialout maxwait** command sets the maximum number of seconds a dial-out record must wait before it can be used to place another call. Setting a dial-out's maximum wait time to its default value of 0 disables the wait time function.

This command changes the maximum wait time for all dial-out records with the same name as the <dialout> parameter.

dialout minwait <dialout> <seconds>

The **dialout minwait** command sets the minimum number of seconds a dial-out record must wait before it can be used to place another call. Be sure to set a dial-out's minimum wait time to a value less than or equal to the dial-out's maximum wait time.

This command changes the minimum wait time for all dial-out records with the same name as the <dialout> parameter.

dialout password <dialout> [<remote-password>]

The **dialout password** command sets or changes the password that is sent to the remote system while attempting to establish a connection with the destination specified by the <dialout> parameter. The remote NetBlazer must have a user ID and password in its user records that match the uname and password parameters of the dial-out record.

This command changes the password for all dial-out records with the same name as the <dialout> parameter. If no <remote-password> is entered, the password is removed from the dial-out record(s).

dialout phone <dialout> [#<item-number>] <phone-number>

The **dialout phone** command sets or changes the telephone number of a dial-out destination.

The <dialout> and optional #<item-number> parameters identify the dial-out record to modify.

The <phone-number> parameter is the new telephone number assigned to the selected dial-out destination.

dialout timeout <dialout> <timeout-minutes>

The **dialout timeout** command sets the amount of time in minutes that a NetBlazer maintains an inactive character mode connection to the dial-out destination specified by the <dialout> parameter.

Note: *The NetBlazer uses different timeout values for character mode and packet mode connections. This command sets the timeout for a character mode connection. The* ***interface dynamic timeout*** *command sets the timeout for a packet mode connection.*

The <timeout-minutes> parameter determines the number of minutes that an inactive character mode connection is maintained. When a line is inactive for the number of minutes specified by the <timeout-minutes> value, the NetBlazer drops the telephone connection. If you set the <timeout> to zero, the connection will never time out.

dialout uname <dialout> [<remote-uname>]

The **dialout uname** command sets or changes the user ID that is sent to the remote system while attempting to establish a connection with the destination specified by the <dialout> parameter. The remote NetBlazer must have a user ID in its user records that matches the uname parameter of the dial-out record.

This command changes the uname value for all dial-out records with the same name as the <dialout> parameter. If no <remote-uname> is entered, the uname value is removed from the dial-out record(s).

Disk Commands

The **disk** command takes you to the Disk command level. To view a list of the commands available at this level, type a question mark (?) at the Disk command level.

```
NB:Top> disk
NB:Top>Disk> ?
Available commands:
chmod           chkdsk          copy            delete
dir             diskcopy        format          loadmodemcap
rename          top>            up>             ?
```

Refer to the *NetBlazer Configuration Guide* for a description of the standard files used by the NetBlazer.

In addition to the standard NetBlazer files, you also can put source files on your diskette. These files contain a series of NetBlazer commands that are run in sequence by the **source** <command-file> command. If you are familiar with the UNIX ed and ex line editors, you can use the NetBlazer's **edit** command to create a source command file.

Although the NetBlazer has its own operating system, the NetBlazer disk format is compatible with MS-DOS. Therefore, as an alternative to using the NetBlazer's **disk** commands, you can use an MS-DOS system that supports 3.5-inch high-density diskettes to perform the following disk maintenance operations:

- Format a diskette
- Copy a NetBlazer diskette
- Copy, rename, or delete files
- Edit the STARTUP.CNF, MODEMCAP.TXT, CHAT.TXT, and PASSWD files, or create command files using a text editor

You can configure the NetBlazer to boot over a TCP/IP network the same way that a diskless workstation boots from a file server on the network. This lets you to keep a single copy of the NetBlazer software and configuration files at a central location.

An alternative to using the BOOTP facilities is to put some or all of the NetBlazer configuration files on a UNIX system and use TFTP to access these files. To do this, you must put a FILES.CNF file on the NetBlazer system diskette.

Refer to the Network Management chapter in the *NetBlazer Configuration Guide* for instructions on configuring the NetBlazer to use remote boot and configuration files.

disk chmod <mode|+w|-w> <filename>

The **disk chmod** command sets the specified files write protection flag. Entering a **chmod -w** <filename> or **chmod 1** <filename> command marks the file as read-only.

```
NB:Top>Disk> chmod -w default.cnf
```

Entering a **chmod +w** <filename> or **chmod 2** <filename> command turns write-protection off for the specified file.

```
NB:Top>Disk> chmod +w default.cnf
```

disk chkdsk [verbose] [fix] [change]

The **disk chkdsk** command displays a status report on the condition of the diskette. Entering this command without any parameters produces a report similar to the following.

```
NB:Top>Disk> chkdsk
2847 fat entries, 9 blocks of fat, 4608 bytes of fat, first fat at 1
14 sectors of root directory starting at 19
2 files, 0 directories, using 78 clusters
2769 free clusters
20 clusters lost (use fix to reclaim)
```

For a more detailed report, enter a **chkdsk verbose** command. For even more detailed information, enter a **chkdsk verbose verbose** command.

Enter a **chkdsk fix** command to attempt to repair a damaged disk.

```
NB:Top>Disk> chkdsk fix
2847 fat entries, 9 blocks of fat, 4608 bytes of fat, first fat at 1
14 sectors of root directory starting at 19
2 files, 0 directories, using 78 clusters
2769 free clusters
20 clusters reclaimed
```

Enter a **chkdsk change** [verbose] [fix] command to check or repair another disk.

```
NB:Top>Disk> chkdsk change fix
Insert diskette and press return
2847 fat entries, 9 blocks of fat, 4608 bytes of fat, first fat at 1
14 sectors of root directory starting at 19
2 files, 0 directories, using 78 clusters
2769 free clusters
20 clusters reclaimed
Insert system diskette and press return
```

disk copy <source-file> <destination-file>

The **disk copy** command copies the file specified by the <source-file> parameter to a file specified by the <destination-file> parameter.

```
NB:Top>Disk> copy startup.cnf startup.sav
Copied 8779 bytes
```

The destination file name must conform to DOS file naming conventions – eight alphanumeric characters with a 3-character extension. Do not use control characters or special symbol such as: *?:;,=+<>|\"[]

disk delete <filename>

The **disk delete** command deletes the file specified by the <file-name> parameter from the diskette.

```
NB:Top>Disk> delete startup.sav
Delete 'startup.sav' [n]? y
```

disk dir [<filename.ext>]

The **disk dir** command displays a list of files on the NetBlazer system diskette. You can display all files on the diskette, a single file referenced by name, or a selected group of files using the * wildcard character to represent an arbitrary string of characters. Several examples are shown below.

Entering a **disk dir** command without specifying a file name displays a list of all files on the diskette.

```
NB:Top>Disk> dir
Directory of A:/

TELEBIT  SYS     14848  02-12-93   11:13
FRED            699280  02-12-93   11:14
BOOTP            24892  01-18-93   14:07
ATK      LDR    152964  02-12-93   11:16
EDIT     LDR     17516  02-12-93   11:16
IPX      LDR    125564  02-12-93   11:16
SNMP     LDR    102516  02-12-93   11:16
TCL      LDR     68856  02-12-93   11:17
CHAT     TXT      3548  02-12-93   11:20
MODEMCAP TXT     22331  02-12-93   11:19
DEFAULT  CNF      1671  02-12-93   11:20
STARTUP  CNF      8779  02-14-93   15:49
PASSWD             138  02-14-93   10:43
13 File(s) (1242903 bytes) 211456 bytes free
```

The next example illustrates how to use the wildcard character (*) to get a list of files that begin with a specified string.

```
NB:Top>Disk> dir pas*
Directory of A:/

PASSWD              138  02-14-93   10:43
1 File(s) (138 bytes) 211456 bytes free
```

The final example shows how to use the wildcard character (*) to produce a directory listing of all files with a common extension.

```
NB:Top>Disk> dir *.cnf
Directory of A:/

DEFAULT  CNF       1671  02-12-93   11:20
STARTUP  CNF       8779  02-14-93   15:49
2 File(s) (10450 bytes) 211456 bytes free
```

disk diskcopy

The **disk diskcopy** command makes a copy of a NetBlazer diskette. The destination diskette must be a formatted 3.5-inch high-density diskette.

Caution:

1. *Any files on the destination diskette are overwritten.*
2. *The **diskcopy** routine does not flush the typeahead buffer – do not press the return key until you insert the appropriate diskette.*

The **diskcopy** routine tells you when to insert the appropriate diskette as shown in the following example.

```
NB:Top>Disk> diskcopy
There will be 3 copy segments (960 blocks each, 2880 blocks total)
0% done...Insert source diskette and press return
33% done reading
Insert destination diskette and press return

33% done writing
33% done...Insert source diskette and press return
66% done reading
Insert destination diskette and press return

66% done writing
66% done...Insert source diskette and press return
100% done reading
Insert destination diskette and press return

100% done writing
Disk copy complete
Insert system diskette and press return
```

disk format [system | /s]

The **disk format** command formats a blank 3.5-inch high-density diskette for use with the NetBlazer.

If you want to initialize the diskette as a system diskette, make sure that you have a system diskette that includes the TELEBIT.SYS file inserted the drive, then enter a **format /s** or **format system** command.

```
NB:Top>Disk> format /s
Please leave system disk in drive until prompted to change it.
Format disk and install system [n]? y
Insert diskette to format and press return

Formatting cylinder 79
Verifying cylinder 79
No bad sectors on disk
writing primary bootstrap to block zero
writing 14848 bytes of secondary bootstrap (telebit.sys)
Format another disk [n]?
Insert system diskette and press return
```

Note: *If you attempt to format a diskette without a system diskette in the drive or enter a **format** command without the system or /s parameter, the NetBlazer asks if you want to format the disk without a system (TELEBIT.SYS file).*

disk loadmodemcap

The **disk loadmodemcap** command reloads the MODEMCAP.TXT file without rebooting the NetBlazer.

```
NB:Top>Disk> loadmodemcap
loading modem information from modemcap.txt
Started 120 second timer to free old modemcap; modemcap locked until then
```

disk rename <oldfile> <newfile>

The **disk rename** command renames the file specified by the <oldfile> parameter to the file specified by the <newfile> parameter.

```
NB:Top>Disk> rename startup.sav startup.bak
```

The new file name must conform to DOS file naming conventions – eight alphanumeric characters with a 3-character extension. Do not use control characters or special symbol such as: *?:;,=+<>|\"[]

Files Command

The **files** [<type>] [[<host>:]<filename>] command specifies the location of the NetBlazer system and configuration files. This allows you to keep some or all of the NetBlazer standard files on a remote TCP/IP system.

Refer to the Network Management chapter in the *NetBlazer Configuration Guide* for instructions on configuring the NetBlazer to use remote boot and configuration files.

Entering a **files** command without any parameter shows the location for all of the NetBlazer system and configuration files.

```
NB:Top> files
files          files.cnf
startup        startup.cnf
passwd         passwd
chat           chat.txt
modemcap       modemcap.txt
default        default.cnf
token          token002.mcd
overlay
bootfile       fred
```

Entering a **files** <type> command shows the location of the file specified by the <type> parameter.

```
pearl:Top> files startup
startup        143.191.10.100:/home/NB/pearl/startup.cnf
```

Initially, all NetBlazer system and configuration files are located on the NetBlazer system diskette. However, you can arrange to have the NetBlazer use BOOTP and/or TFTP to access NetBlazer system and configuration files on a remote TCP/IP host. The following display shows the contents of the FILES.CNF file for a NetBlazer that is configured to get its files from a TCP/IP host on the network.

```
configure file startup 143.191.10.100:/home/NB/pearl/startup.cnf
configure file passwd 143.191.10.100:/home/NB/pearl/passwd
configure file modem 143.191.10.100:/home/NB/modemcap.txt
configure file chat 143.191.10.100:/home/NB/chat.txt
configure file default 143.191.10.100:/home/NB/default.cnf
configure file token 143.191.10.100:/home/NB/token002.mcd
configure file overlay 143.191.10.100:/home/NB
```

If you have several NetBlazers on your network, you can use the %s format to indicate the host name of the NetBlazer. For example, the %s string in the following display tells the NetBlazer to substittute its host name in place of the %s string.

```
configure file startup 143.191.10.100:/home/NB/%s/startup.cnf
configure file passwd 143.191.10.100:/home/NB/%s/passwd
configure file modem 143.191.10.100:/home/NB/modemcap.txt
configure file chat 143.191.10.100:/home/NB/chat.txt
configure file default 143.191.10.100:/home/NB/default.cnf
configure file token 143.191.10.100:/home/NB/token002.mcd
configure file overlay 143.191.10.100:/home/NB
```

This allows you to use a single FILES.CNF file instead of one per NetBlazer. A NetBlazer named pearl gets its startup and password information from the files in the /home/NB/pearl directory of the host at 143.191.10.100; where as a NetBlazer named jade gets its startup and password information from the /home/NB/jade directory of the same host.

Group Command

The **group** command takes you to the Configure Group command level. To view a list of the commands available at this level, type a question mark (?) at the Configure Group command level.

```
NB:Top> group
NB:Top>Configure>Group> ?
Available commands:
add             delete          list            modify
top>            up>             ?
```

These commands set up and maintain line characteristics groups. Line characteristics groups are used to select an appropriate asynchronous line when dialing out to a particular destination by comparing the dialout destination characteristics to the characteristics assigned to each line. Only lines that exactly match the dialout destination requirements can be used to call that destination.

group add <group-name>

The **group add** command adds a group name to the line characteristics group list. After adding a new group name to the group list, use the **group modify** command to assign characteristic tags to the group.

group delete <group-name>

The **group delete** command removes the specified group from the line characteristics group list.

group list [<group-name>]

The **group list** command displays the specified group and its assigned characteristics. The optional <group-name> parameter specifies which group to display. If this command is entered without a <group-name> variable, a list of group names is displayed.

The NetBlazer comes with seven standard line characteristic groups: dialout, v32dialout, pepdialout, request, raw, raw_dial, and v25dialout. Refer to the **line characteristics** command description for a description of these characteristic groups.

group modify <group-name> add | delete <string>
group modify <group-name> delete #<string-number>

The **group modify** command adds or deletes characteristic tags assigned to a line characteristics group.

The <group-name> parameter is the name of the group previously created through a **group add** command.

When adding a characteristic tag to a group, the <string> parameter is the characteristic tag assigned to the specified group. The characteristic tag can be a user defined string or a preset characteristic tag that is defined in the MODEMCAP.TXT file.

When deleting a characteristic from a group using the first form of the command, the <string> parameter is the characteristic that is to be removed from the specified group. In the second form of the command, the #<string-number> parameter is the number assigned to the characteristic tag. Enter a **group list** <group-name> command to view a string's assigned number.

The following example shows you how to create a group named telebit that includes two characteristics tags, dialout and branch.

```
NB:Top>Configure>Group> add telebit
NB:Top>Configure>Group> modify telebit add dialout
NB:Top>Configure>Group> modify telebit add branch
NB:Top>Configure>Group> list telebit
Group :
    telebit
Characteristics :
    0: dialout
    1: branch
```

The next two examples show you how to remove a characteristic tag from the group telebit using either of the **group modify delete** command formats.

```
NB:Top>Configure>Group> modify telebit delete #0
NB:Top>Configure>Group> modify telebit delete branch
```

History Commands

The **history** command takes you to the History command level. To view a list of the commands available at this level, type a question mark (?) at the History command level.

```
NB:Top> history
NB:Top>History> ?
Available commands:
?               list            status          top>
on              off             prompt          up>
```

These commands control the NetBlazer history mechanism, which saves a list of recent NetBlazer commands. You can use this list to avoid repetitive typing by using a shorthand notation to enter commands based on previous commands.

The following table summarizes the syntax of the history commands.

Command Syntax	Description
history list [<number-of-lines>]	Displays a list of recent commands (events).
history off	Turns off the history mechanism and clears the history list.
history on [<number-of-lines>]	Turns on the history mechanism and optionally sets the maximum number of lines in the history list.
history prompt	Displays an event number at the end of each command prompt when history is on.
history status	Displays the current settings of the history mechanism.
!!	References the last command entered. By itself, repeats the last command entered.
!<event-number>	Invokes the command specified by the event number.
!<string>	Invokes the most recent command beginning with <string>.
!?<string>**?**	Invokes the most recent command containing <string>.
!<event>**:p**	Displays and makes the specified event current but does not invoke the command.
![<event>]:<number>	References field <number> of the specified event. If no <event> is specified, references the field <number> of the last command.
![<event>]:<start>–<end>	References fields ranging from <start> to <end> of specified event.
![<event>]:$	References the last field of the specified event. To reference the last field of the previous command, enter **!$**.
^<old string>^<new string>^	Replaces <old string> with <new string> in last command.

Referencing Previous Commands

The history mechanism uses the exclamation mark (!) followed by an event variable to reference a previous command. The event variable may be a second exclamation mark (!) to reference the previous command, a number to reference a recent command by its event number, or a character string to reference the most recent command that begins with the character string.

Note: *To prevent the history mechanism from misinterpreting an exclamation mark (!) used in other NetBlazer commands when history is on, you must escape the exclamation mark by preceding it with a backslash (\). For example, !23 in a filter command should be enter as \!23 when history is on.*

Repeating Previous Commands

The next several examples show you how to use the history mechanism based on the following list of recent commands.

```
NB:Top> [4] history list
    1 history prompt
    2 escape
    3 ping pearl 1024 3 0 1
    4 history list
```

To reissue the last command (**history list** in the previous example), enter **!!** at the command prompt.

```
NB:Top> [5] !!
history list
    1 history prompt
    2 escape
    3 ping pearl 1024 3 0 1
    4 history list
    5 history list
```

To reissue some other recent command, enter **!** followed by the event number of the command.

```
NB:Top> [6] !3
ping pearl 1024 3 0 1
PING pearl(143.191.63.1): 1024 bytes      min/avg/max/mdev
Type ^] (decimal 29) <CR> to return to NetBlazer
1024 bytes from 143.191.63.1 icmp_seq=0 time=15 ms  15/15/15/0
1024 bytes from 143.191.63.1 icmp_seq=1 time=10 ms  10/14/15/1
1024 bytes from 143.191.63.1 icmp_seq=2 time=10 ms  10/14/15/2
----pearl PING Statistics----
3 packets transmitted, 3 packets received, 0% packet loss
round-trip (ms)  min/avg/max/mdev = 10/14/15/2
```

You can repeat a command by matching the first word or part of the first word in the command string. The following example shows you how to reissue the last **escape** command entered.

```
NB:Top> [7] !e
escape
Current escape character is ^] (decimal 29)
```

Entering **!e** repeats the most recent command that starts with the letter **e**. If the history list contains other commands that begin with **e**, you may need to enter additional characters to distinguish one command from another.

You also can repeat a command by matching a character string contained in the command string. The next example shows you how to reissue the last command that contained the word pearl.

```
NB:Top> [8] !?pearl?
ping pearl 1024 3 0 1
PING pearl(143.191.63.1): 1024 bytes      min/avg/max/mdev
Type ^] (decimal 29) <CR> to return to NetBlazer
1024 bytes from 143.191.63.1 icmp_seq=0 time=10 ms  10/10/10/0
1024 bytes from 143.191.63.1 icmp_seq=1 time=10 ms  10/10/10/0
1024 bytes from 143.191.63.1 icmp_seq=2 time=10 ms  10/10/10/0
----pearl PING Statistics----
3 packets transmitted, 3 packets received, 0% packet loss
round-trip (ms)  min/avg/max/mdev = 10/10/10/0
```

Notice that you must surround the reference string with question marks (?) to reference the last command containing the string.

Building New Commands from Previous Commands

You can build a new command by extending a previous command, as illustrated below.

```
NB:Top> [9] !e ctrl-z
escape
Current escape character is ^Z (decimal 26)
NB:Top> [10] !!:0
escape
Current escape character is ^Z (decimal 26)
```

In the above example, we added a ctrl-z to the end of the most recent **escape** command, thereby changing the escape character to ^Z (decimal 26).

On line 10, we confirmed that the escape character was changed by directing the NetBlazer to issue a command that consisted of the first word (field 0) of the last command in the history list. This illustrates how you can build new commands by using selected fields from previous commands.

The history mechanism treats each command line as a series of numbered fields starting at 0. The fields are separated by spaces. For example, the **route add** command shown below consists of six fields numbered 0-5.

```
0    1        2        3        4        5
route add 143.191.13/24 en0 143.191.10.29 2
```

You can refer to individual fields in a command by entering :<field-number> after the event identification. The :<field-number> identifies which field to use in the new command. For example, assume that you have just added a route, and you want to verify that the route is correct by doing a route lookup.

```
NB:Top>Configure>IP> [14] route add 143.191.13/24 en0 143.191.10.29 2
NB:Top>Configure>IP> [15] !!:0 lookup !!:2
route lookup 143.191.13/24
Dest            Len Interface    Gateway          Metric  P Timer  Use
143.191.13.0    24  en0          143.191.10.29    2         0      0
```

The **!!:0** and **!!:2** in line 15 refer to the first and third field of the previous command.

You can also specify a range of fields to use in the new command. For example, if you want to enter a second route that uses the same interface and gateway as the previous **route add** command, you could enter the following command.

```
NB:Top>Configure>IP> [16] !14:0-1 143.191.14/24 !14:3-5
route add 143.191.14/24 en0 143.191.10.29 2
```

Notice how this command uses the range of fields from history line 14 to build the new **route add** command.

Editing Previous Commands

You can build new commands by editing the previous command. This is useful for correcting typing errors or changing words within the command. The command syntax for editing the previous command is:

^old string^new string^

The caret symbol (^) is used to separate the text that you want to replace from the new text. The rest of command line remains unchanged.

To illustrate, assume that you want to **ping** another host using the same parameters of the last **ping** command issued. First, make the last **ping** command current.

```
NB:Top>Configure>IP> [17] !ping:p
ping pearl 1024 3 0 1
```

The **:p** in the above command makes the command specified by **!ping** current but does not execute the command.

Next, enter the following command to change the host name **pearl** to the new host name **napa**.

```
NB:Top>Configure>IP> [18] ^pearl^napa^
ping napa 1024 3 0 1
PING napa(143.191.10.100): 1024 bytes      min/avg/max/mdev
Type ^Z (decimal 26) <CR> to return to NetBlazer
1024 bytes from 143.191.10.100 icmp_seq=0 time=15 ms  15/15/15/0
1024 bytes from 143.191.10.100 icmp_seq=1 time=20 ms  15/16/20/1
1024 bytes from 143.191.10.100 icmp_seq=2 time=20 ms  15/17/20/2
----napa PING Statistics----
3 packets transmitted, 3 packets received, 0% packet loss
round-trip (ms)  min/avg/max/mdev = 15/17/20/2
```

Notice that the NetBlazer issued a **ping** command to **napa** using the same parameters that it had previously used to **ping pearl**.

history list [<number-of-lines>]

The **history list** command displays a list of recent commands (events).

```
NB:Top> [5] history list
    1 history status
    2 history prompt
    3 escape
    4 ping pearl 1024 3 0 1
    5 history list
```

The above example shows five commands, which were entered since the last time history was turned on.

You can display just part of the history list by entering the number of lines to display.

```
NB:Top> [6] history list 4
    3 escape
    4 ping pearl 1024 3 0 1
    5 history list
    6 history list 4
```

This example displays the last four commands in the history list.

Note: *If you do not specify the number of lines to display, the NetBlazer displays up to 20 lines. You can display more than 20 lines by specifying a number greater than 20 lines.*

history off

The **history off** command turns off the history mechanism and clears the list of recent commands.

```
NB:Top> history off
NB:Top> history status
history not on
```

history on [<number-of-lines>]

The **history on** command turns on the history mechanism and optionally sets the maximum number of lines in the list.

```
NB:Top> history on 10
NB:Top> history status
History is on: max=10 count=1 first=1 prompt=off debug=0
```

If you do not specify the number of lines in the **history on** command, the NetBlazer uses the current maximum.

history prompt

The **history prompt** command instructs the NetBlazer to display an event number at the end of each command prompt when history is on.

```
NB:Top> history prompt
NB:Top> [3] escape
Current escape character is ^Z (decimal 26)
NB:Top> [4]
```

Notice that the event numbers are incremented with each command. These event numbers are used to repeat or modify a recent command referenced by its event number as described in the following **history list** command.

history status

The **history status** command displays the current settings of the history mechanism.

```
NB:Top> history status
history is on: max=20 count=4 first=1 prompt=off debug=0
```

The previous example shows the NetBlazer's default history settings. The NetBlazer is shipped with history turned on. The maximum number of history lines displayed by the **history list** command is 20, and the **history prompt** is turned off.

The count field shows the current number of entries in the history list, and the first field shows the number of the first entry in the list. The debug field is used for maintenance purposes only.

Hostname Command

The **hostname** [<hostname>] command sets or displays the hostname of the NetBlazer which appears in the NetBlazer command prompt. If the <hostname> parameter is omitted in the command, the current hostname is displayed.

Note: *The hostname is used by the **pppdial** and **slipdial** routines to create a dynamic dial-up interface with a remote NetBlazer.*

Interface Commands

The **interface** command takes you to the Configure Interface command level. To view a list of the commands available at this level, type a question mark (?) at the Configure Interface command level.

```
NB:Top> interface
NB:Top>Configure>Interface> ?
Available commands:
attach          backup          delete          dump-mib
dynamic>        list            mdelay          mru
mtu             rif             start           stop
top>            trace           tracedump       tracelimit
up>             ?
```

Commands at this level configure and control the interfaces used by the NetBlazer.

The NetBlazer uses three types of interfaces: LAN, dynamic, and dedicated.

LAN interfaces are automatically created for each Ethernet and Token Ring board installed in the NetBlazer. Before using these interfaces, you must configure the interfaces as described in the *NetBlazer Configuration Guide*.

Dynamic interfaces provide dial-up links to route packets over the public switched telephone network. Refer to the **interface dynamic** command descriptions for instructions on setting and maintaining dynamic interfaces.

Dedicated interfaces route packets over dedicated point-to-point serial links. Use the **interface attach** command to attach a dedicated interface to a synchronous or asynchronous serial line.

You can start a **watchdog** server to monitor the state of a dedicated interface. If the dedicated interface link fails, the NetBlazer automatically reroutes the link's traffic through a dynamic backup interface. A backup interface is assigned to a dedicated interface through the **interface backup** command.

interface attach <line-name> slip|ppp <interface> [<mru>] [<ip-addr>]

The **interface attach** command configures a synchronous or asynchronous serial line for routing packets over a dedicated PPP or SLIP interface. As illustrated in Figure INT-1, a dedicated PPP or SLIP interface may be used to interconnect two NetBlazers or to attach a workstation that runs PPP or SLIP software to the NetBlazer.

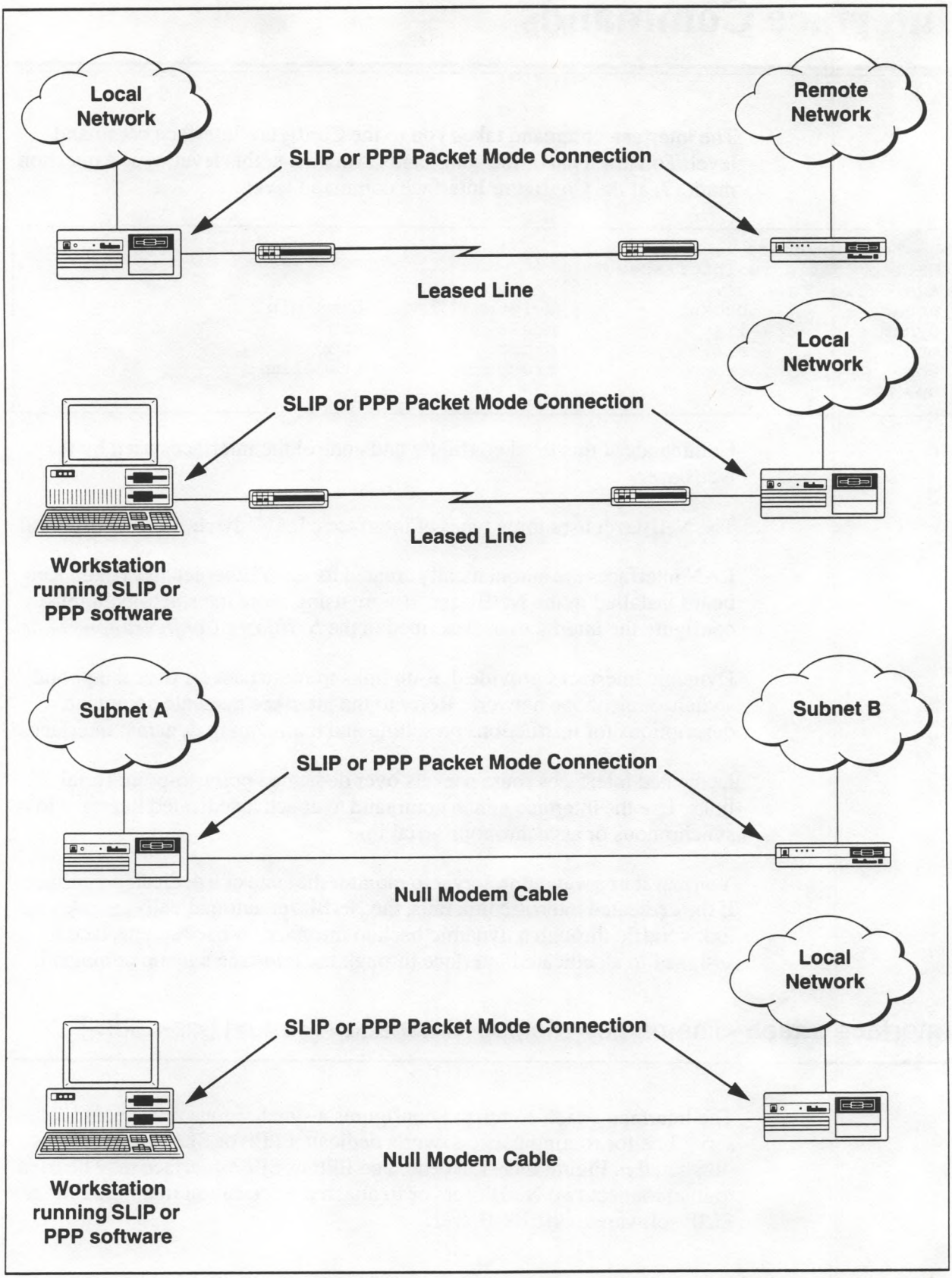

Figure INT-1. Dedicated SLIP or PPP Links

Note: *Creating a dedicated interface automatically turns off* ***login*** *for the line used by that interface.*

The <line-name> parameter specifies which serial line to use as a dedicated SLIP or PPP link.

The slip | ppp parameter determines the framing protocol used to transmit packets over the serial link. Synchronous lines must use PPP framing. If the interface is used for transferring only IP packets over an asynchronous link, you may choose SLIP framing. If the interface is used to transfer Appletalk or IPX packets, use PPP framing.

The <interface> parameter is the name that you assign to the new interface created by this command.

The optional <mru> (maximum receive unit) parameter sets the number of bytes in the largest packet that may be received over the dedicated link. This parameter defaults to 1006 for a SLIP link and 1500 for a PPP link.

The optional <IP-address> parameter assigns an IP address to the named interface. If this parameter is not entered, the NetBlazer global IP address is assigned to the interface.

The following example illustrates how to attach a dedicated PPP interface, named syn-opal, to the syn0 synchronous line.

```
pearl:Top>Configure>Interface> attach syn0 ppp syn-opal
syn0: interface=syn-opal, protocol=PPP, MRU=1500, ip address=143.191.63.1
```

Before you can actually use this dedicated PPP link, you must configure the NetBlazer to route packets through this interface. You also must repeat this procedure on the other side of the link. Refer to the *NetBlazer Configuration Guide* for additional instructions on setting up a dedicated link.

interface backup <interface> [<backup-interface>]

The **interface backup** command selects a dynamic interface to back up a dedicated serial line interface where <interface> is the name of the interface, and <backup-interface> is the name of the dynamic interface to use when the **watchdog** function determines that the interface line is down.

When the interface goes down and the NetBlazer switches to the backup interface, all routes using the interface are changed to the backup interface. When the interface line returns, the routes through the original interface are returned.

Entering an **interface backup** <interface> command without specifying the <backup-interface> name turns off backup for the specified interface.

interface delete <interface>

The **interface delete** command deletes a specified dynamic or dedicated interface from the interface list. You cannot delete the loopback, null, error, or static interfaces.

Note: *Any user, dialout destination, and routes associated with a dynamic interface are also deleted.*

*The **line login** for a attached line is not turned back on. You must turn login on if you want the login prompt to be issued when a connection is made to the line.*

interface dump-mib <interface>

The **interface dump-mip** command displays the variables maintained by each interface for SNMP as defined in RFC-1213.

interface dynamic

The **interface dynamic** command takes you to the Configure Interface Dynamic command level. To view a list of the **dynamic** commands, type a question mark (?) at the Configure Interface Dynamic command level.

```
NB:Top>Configure>Interface> dynamic
NB:Top>Configure>Interface>Dynamic> ?
Available commands:
add             auto-arp        auto-route      delete
farside         framing         ipdial          list
pppdial         prompt          queue-max       slipdial
timeout         top>            up>             ?
```

Commands at this level set up and maintain dynamic interfaces, which are used to route packets over the public telephone network.

Note: *The preferred method for creating a dynamic interface is through the **ipdial**, **slipdial**, or **pppdial** commands. Using these commands ensures that all of the information needed to set up a dynamic interface is provided.*

*If you decide to use the **dynamic add** command instead, you must create a dialout and/or user record with the same name as the interface. This allows the NetBlazer to establish outbound and/or inbound connections through the interface. You also must add one or more routes to the routing table to send packets destined for the remote site through the interface.*

Adding ARA users through the **user add** command creates a dynamic interface for ARA connections.

Refer to the *NetBlazer Configuration Guide* for detailed instructions on how to set dynamic interfaces.

interface dynamic add [<interface>] [<uname>] [<password>] [<crypto>] [<chat>] [slip|ppp|ara] [<mru>]

The **dynamic add** command creates a new dynamic interface for a dial-up connection to a remote site. Each parameter in this command must be entered in sequence. Enter 0 as a placeholder for parameters that you want set to <<NONE>>. If you omit any of the parameters, you are prompted for the remaining parameters.

Note: *Before adding a dynamic interface, first create a user record and/or dialout record with the same name as the interface. Refer to the* ***user add*** *and* ***dialout add*** *command descriptions for instructions.*

The <uname>, <password>, <crypto> and <chat> parameters are assigned to a dialout record with the same name as the <interface>. If the dialout record does not exist, these parameters are set to <<NONE>>.

The <interface> parameter is the name assigned to the interface.To allow inbound or outbound connections through this interface, you must set up a user record and/or dialout record with the same name as the interface.

The <uname> parameter is the user ID that is sent in response to a login prompt from the remote NetBlazer. The <password> parameter is sent in response to a password prompt, and the <crypto-key> parameter is used to encrypt a crypto response when a crypto challenge is received from the remote site.

To authenticate a dial-out connection, the remote NetBlazer must have a user record that matches the interface's <uname>, <password>, and <crypto-key> parameters.

The <chat> parameter specifies which chat script in the CHAT.TXT file to use when attempting to establish a dial-out connection with the remote site. Use the ics chat script for interfaces used to dial out to another NetBlazer.

The slip|ppp|ara parameter specifies which protocol to use when sending packets through this interface. If the interface is used to transfer IPX or AppleTalk packets, use PPP. If the interface is used for an ARA connection, choose ARA. If the interface is used to transfer only IP packets, you may use either PPP or SLIP.

The <mru> parameter sets the number of bytes in the largest packet that may be received or transmitted through this interface. We recommend that you use the default MRU/MTU setting: 1500 for PPP or ARA connections, 1006 for SLIP connections.

interface dynamic auto-arp <interface> <add|delete> <ip-addr ...>

The **dynamic auto-arp** command configures the NetBlazer to automatically create proxy ARPs and related routes for one or more host IP addresses when a connection is established through a specified dynamic interface. This accommodates users who temporarily need to move their system from the local network to a remote site.

Note: *Using the auto-arp function limits establishment of a dynamic connection to the inbound direction only. In other words, hosts on the local side cannot communicate with the remote site through this interface until a connection is established from the remote site.*

Refer to the *NetBlazer Configuration Guide* for an example of how the **dynamic auto-arp** command is used.

The **dynamic auto-arp** <interface> **add** <ip-addr ...> command adds host IP addresses to the list of addresses for which the NetBlazer creates proxy ARPs and related routes when the specified dynamic interface is connected. The auto-ARP entries (shown as **published,auto** in the ARP list) and related routes (marked by an A in the P field of the route list) only remain while the connection is present. When the connection is dropped, the auto-ARP entries and routes are removed.

The **dynamic auto-arp** <interface> **delete** <ip-addr ...> command removes host IP addresses from the list of addresses for which the NetBlazer creates proxy ARPs and related routes when a dynamic connection is made.

interface dynamic auto-route <interface> <add|delete> <ip-addr[/bits] ...>

The **dynamic auto-route** command configures the NetBlazer to automatically add one or more host or network IP addresses to the routing table when a connection is established through a specified dynamic interface. When the connection is dropped, the dynamic interface's routes, marked by the letter A in the P field of the route list, are removed. This reduces the number of entries in the routing table by only listing routes through a dynamic interface when it is in use.

Note: *Using the auto-route function limits establishment of a dynamic connection to the inbound direction only. In other words, hosts on the local side cannot communicate with the remote site through this interface until a connection is established from the remote site.*

The **dynamic auto-route** <interface> **add** <ip-addr[/bits] ...> command adds IP addresses to the list of auto-route addresses for a specified interface.

The **dynamic auto-route** <interface> **delete** <ip-addr[/bits] ...> command removes IP addresses from the auto-route address list for the specified interface.

Refer to the *NetBlazer Configuration Guide* for an example of how the **dynamic auto-route** command is used.

interface dynamic delete <interface>

The **dynamic delete** command removes a specified dynamic interface from the interface list.

Note: *Deleting a dynamic interface also removes any user and/or dialout record with the same name as the interface and purges the routing table of any routes through that interface.*

interface dynamic farside <interface>

The **dynamic farside** command displays information describing how a connection to the remote site is established through this interface. This information is also displayed at the end of an **ipdial** routine.

interface dynamic framing <interface> slip|ppp

The **dynamic framing** command selects the framing method used to transfer packets through a specified dynamic interface. The same framing method must be used on both sides of a dynamic link. If the interface is used for transferring only IP packets, choose SLIP framing. If the interface is used to transfer AppleTalk or IPX packets, use PPP framing.

interface dynamic ipdial

The **dynamic ipdial** command is one of three routines that step you through the process of creating a dynamic dial-up interface. The **ipdial** routine is the best choice for setting up multi-line connections to the same destination. If you are setting up a single line interface, consider using **slipdial** or **pppdial** instead. These two routines simplify the process by using the NetBlazer's defaults whenever possible.

You need the following information to set up a dynamic interface using **ipdial**:

- The user IDs, passwords, and crypto keys used to authenticate an inbound and/or outbound connection.

- The telephone number(s) used to call the remote NetBlazer.
- The serial link framing protocol (SLIP or PPP) used to encapsulate IP packets sent between the two sites. Both sites must use the same framing method. If the interface is used for transferring only IP packets, choose SLIP framing. If the interface is used to transfer AppleTalk or IPX packets, use PPP framing.
- The network address of each remote LAN segment that can be reached either directly or indirectly through this interface.

The same information is required at the remote site to set up the dynamic interface on the other side of the link.

Refer to the *NetBlazer Configuration Guide* for detailed instructions on how to set up dynamic interfaces.

interface dynamic list [<interface>]

The **dynamic list** command displays status information about the dynamic interfaces that are currently used by the NetBlazer.

If the optional <interface> parameter is entered, additional information for the named interface is displayed. This information varies depending on the type of interface and its present status.

interface dynamic pppdial [<interface>]

The **dynamic pppdial** command starts a dialog that steps you through the process of setting up a dynamic PPP interface to a remote site. In order to keep the dialog as simple as possible, the NetBlazer's defaults are used whenever possible.

To set up a dynamic interface using **pppdial**, you need the following information:

- The host name and telephone number of the remote NetBlazer.
- A crypto key or password used to authenticate the connection between the local and remote site.
- The network address of each remote LAN segment that can be reached through this dynamic interface.

The same information is required at the remote site to set up the dynamic interface on the other side of the link.

If you enter the optional <interface> parameter, the NetBlazer will not prompt you for the name of the remote system in the **pppdial** dialog.

Note: *When using **pppdial** to set up a dynamic interface between two NetBlazers, be sure to use the hostname of the remote NetBlazer as the interface name. Also, be sure to use the same crypto key or password for both interfaces.*

An example of the **pppdial** dialog is shown in the following display.

```
jade:Top>Configure>Interface>Dynamic> pppdial
Name of the remote (other) system: ruby
Phone number: 9-555-1234
Authentication method (crypto, password, none) [c]:
Enter the crypto key (no echo):
Re-enter the crypto key:
Add dialout 'ruby'
Add user 'ruby'
Add dynamic interface=ruby, user=jade chat=ics, with crypto handshaking
 using PPP, MRU=1500
Okay (yes|no|quit) [y]?
REMINDER! You need to set up route(s) for this interface
```

This dialog creates an interface, dialout, and user record with the same name as the remote system (ruby in the above example).

The interface record is configured to use PPP framing and an MRU/MTU of 1500 by default.

After creating the dynamic interface to the remote site, you must add the interface to the IPX and/or AppleTalk interface list (if applicable) and set up static routes through this interface for each LAN segment on the remote network.

This process of creating an interface and setting up routes must be repeated at the remote site. If the remote NetBlazer is installed and ready to accept calls, you can use the **dial** command to call the remote NetBlazer and run **pppdial** from your local NetBlazer.

interface dynamic prompt <interface> [<text-string>]

The **dynamic prompt** command sets the text string that is sent by NetBlazer when a packet mode connection is successfully established through the specified interface. Be sure to enclose <text-strings> that include spaces in quotes.

If you do not enter a <text-strings>, the NetBlazer displays the interface's current prompt setting.

For NetBlazer to NetBlazer connections, use the default prompt setting – Packet mode enabled.

interface dynamic queue-max <interface> <queue-length>

The **dynamic queue-max** command sets the maximum number of packets that may be queued for a designated interface.

Once the queue reaches the value specified by the <queue-length> parameter, additional packets are discarded.

interface dynamic slipdial [<interface>]

The **dynamic slipdial** command starts a dialog that steps you through the process of setting up a dynamic SLIP interface to a remote site. In order to keep the dialog as simple as possible, the NetBlazer's defaults are used whenever possible.

To set up a dynamic interface using **slipdial**, you need the following information:

- The host name and telephone number of the remote NetBlazer.
- A crypto key or password used to authenticate the connection between the local and remote site.
- The network address of each remote LAN segment that can be reached through this dynamic interface.

The same information is required at the remote site to set up the dynamic interface on the other side of the link.

If you enter the optional <interface> parameter, the NetBlazer will not prompt you for the name of the remote system in the **slipdial** dialog.

Note: *When using **slipdial** to set up a dynamic interface between two NetBlazers, be sure to use the hostname of the remote NetBlazer as the interface name. Also, be sure to use the same crypto key or password for both interfaces.*

An example of the **slipdial** dialog is shown in the following display.

```
jade:Top>Configure>Interface>Dynamic> slipdial
Name of the remote (other) system: ruby
Phone number: 9-555-1234
Authentication method (crypto, password, none) [c]:
Enter the crypto key (no echo):
Re-enter the crypto key:
Add dialout 'ruby'
Add user 'ruby'
Add dynamic interface=ruby, user=jade chat=ics, with crypto handshaking
 using SLIP, MRU=1006
Okay (yes|no|quit) [y]?
REMINDER! You need to set up route(s) for this interface
```

This dialog creates an interface, dialout, and user record with the same name as the remote system (ruby in the above example).

The interface record is configured to use SLIP framing and an MRU/MTU of 1006 by default.

After creating the dynamic interface to the remote site, you must add the interface to the IPX and/or AppleTalk interface list (if applicable) and set up static routes through this interface for each LAN segment on the remote network.

This process of creating an interface and setting up routes must be repeated at the remote site. If the remote NetBlazer is installed and ready to accept calls, you can use the **dial** command to call the remote NetBlazer and run **slipdial** from your local NetBlazer.

interface dynamic timeout <interface> <timeout-seconds>

The **dynamic timeout** command sets the amount of time in seconds that a NetBlazer maintains an inactive packet mode connection associated with a designated interface.

The NetBlazer starts a timer on each packet mode connection. This timer is reset each time a packet is transmitted or received through the interface. When the interface has been inactive (no packets have been transmitted or received) for the number of seconds specified by the <timeout-secs> value, the NetBlazer drops the connection.

Setting an interface's timeout value to 0 disables the timer on one side of the connection. If you set the timers on both sides of the connection to 0, the connection will never time out. In this case, you must enter a **line reset** command to break the connection.

interface list [<interface>]

The **interface list** command displays status information about the interfaces that are currently used by the NetBlazer.

If the optional <interface> parameter is entered, additional information for the named interface is displayed. This information varies depending on the type of interface and its present status.

interface mdelay <interface> [<seconds>] [<threshold>]

The **interface mdelay** command sets the parameters that the NetBlazer uses to measure the traffic through an interface in order to determine if the other dynamic interfaces in a balanced route should be brought up. Because there is no way to predict which interface in a balanced route will be used first, you should set the **mdelay** value for all interfaces in the group.

If the interface's total traffic divided by its expected capacity for the specified number of seconds exceeds the threshold value, the NetBlazer brings up the remaining dynamic interfaces in the balanced route.

The <seconds> parameter affects the NetBlazer response to instantaneous bursts in the interface's traffic – the lower this parameter is set, the more likely the NetBlazer is to bring up the remaining interfaces in response to a short burst of traffic through the interface. If you set this parameter to a higher value, the NetBlazer is more likely to bring up the other interfaces based on the average level of traffic through the interface.

If the interface's **mdelay** is set to its default value of 0 seconds, all interfaces in the balanced route are brought up immediately.

The <threshold> parameter is a percentage value that sets the trigger point at which the NetBlazer brings up the remaining interfaces. If you set this parameter to a higher value, the NetBlazer is less likely to bring up the other interfaces.

interface mru [<interface>] [<mru>]

The **interface mru** command sets the maximum receive unit (i.e., number of octets or bytes) that may be transmitted through a specified interface.

If this command is entered without the <interface> or <mru> parameters, the system prompts you for the required information.

We recommend that you use the default MRU setting: 1500 for PPP or 1006 for SLIP connections.

interface mtu [<interface>] [<mtu>]

The **interface mtu** command sets the maximum transmit unit (i.e., number of octets or bytes) that may be transmitted through a specified interface.

If this command is entered without the <interface> and <mtu> parameters, the system prompts you for the required information.

You cannot set the MTU for a PPP interface. This value is negotiated based on the peer MRU.

We recommend that you use the default MTU setting of 1006 for SLIP connections.

interface rif <interface> [<node-address>]

The **interface rif** command displays the contents of the Routing Information Fields, which are currently used only in Token Ring environments. For additional information, refer to *RFC 1042, IEEE 802.5 (Token Ring Media Access Control)* or the *IBM Token-Ring Network Architecture Reference Manual.*

interface start <interface>

The **interface start** command restarts an interface that has been previously turned off by a **interface stop** command.

interface stop <interface>

The **interface stop** command turns off a specified interface. This command only works for static or dedicated interfaces. It cannot be used to stop a dynamic interface.

interface trace [<interface>] [<options ...>]

The **interface trace** command allows you to selectively monitor inbound and/or outbound traffic through one or more interfaces. Trace output for all interfaces is sent to only one serial port or telnet session – the last one used to issue a **trace** command. However, you can use the **ofile=**<filename> option to redirect the trace output to a file. For example, **trace en0 on ofile=trace.out** sends the trace output for en0 to the file named TRACE.OUT.

To view a trace file, turn off tracing, then enter a **more** <filename> command.

The <interface> parameter specifies which interface is to be traced. This is the name assigned to a selected interface.

The <options ...> parameters defines the form and content of the trace output. Refer to Table INT-1.

Table INT-1. Interface Trace Options

Option	Description
on, all	Enable input and output packet tracing for all packet types (ip, ipx, appletalk, unknown, compressed, uncompressed, raw).
dis[able], off	Stop tracing specified interface.
ena[ble], resume	Resume tracing specified interface.
clear, none	Clear all tracing option bits of specified interface.
in[put], noin[put]	Enable/disable tracing of inbound packets.
out[put], noout[put]	Enable/disable tracing of outbound packets.
comp[ressed]	Enable tracing of compressed packets.
nocomp[ressed]	Disable tracing of compressed packets.
uncomp[ressed]	Enable tracing of uncompressed packets.
nouncomp[ressed]	Disable tracing of uncompressed packets.
raw	Enable tracing of raw packets.
noraw	Disable tracing of raw packets.
ip, noip	Enable/disable IP packet tracing.
ipx, noipx	Enable/disable IPX packet tracing.
atk, noatk	Enable/disable AppleTalk packet tracing.
appletalk, noappletalk	Same as atk, noatk.
unknown, nounknown	Enable/disable tracing of unknown packet types.
onlyip	Disable packet tracing for all protocols except IP.
onlyipx	Disable packet tracing for all protocols except IPX.
onlyatk	Disable packet tracing for all protocols except AppleTalk.
onlyunknown	Disable packet tracing for all protocols except unknown packet types.
hex	Show packet contents in both HEX and ASCII format.
ascii	Show packet contents in ASCII format only.
header[s]	Only show packet headers. This is the default if HEX or ASCII is not set.

Notes:

Entering a **trace off** command without specifying an interface, stops tracing for all interfaces.

To trace a SLIP interface, you must specify at least one of the comp, uncomp or raw options. These optons are only valid when tracing SLIP interfaces.

Entering an **interface trace** command without any parameters displays the current status of tracing for each interface.

If the trace output falls too far behind (i.e., more than 200 packets in the queue), trace information is discarded to avoid tying up too much memory. If **syslog** is enabled, an informational message is logged indicating how many trace packets were lost.

You can filter trace information using the **debug filter** commands, which work the same as the standard packet filtering commands. For example, if you only want to view trace information on the en0 interface for packets that have a source or destination address of 143.191.64.1, enter the following **debug filter** and **interface trace** commands.

```
NB:Top> debug filter
NB:Top>Configure>Debug>Filter> permit 143.191.64.1 en0
NB:Top>Configure>Debug>Filter> list
Source              Destination         Interface Protocol I/O  Permit/Deny   Use
143.191.64.1/32                         en0                In   Permit          0
                    143.191.64.1/32     en0                In   Permit          0
143.191.64.1/32                         en0                Out  Permit          0
                    143.191.64.1/32     en0                Out  Permit          0
Default                                 en0                     Deny
notify on
NB:Top>Configure>Debug>Filter> interface trace en0 on
en0 enabled compressed uncompressed raw input output status ip ipx appletalk
unknown
Tracing to line line17
```

If you start a trace from a telnet session that uses the same interface being traced, debug filters are automatically set up to filter out the telnet session's packets. An example of the filters that are automatically added to the debug filter list is shown below.

```
NB:Top>Configure>Interface> debug filter list
Source              Destination         Interface Protocol I/O  Permit/Deny   Use
143.191.10.100/32   143.191.11.5/32     en0       TCP      In   Deny           12
                                          port rules: =23
143.191.11.5/32     143.191.10.100/32   en0       TCP      Out  Deny           11
                                          port rules: =4004
Default                                 en0                     Permit
notify on
```

These filters are automatically removed when the telnet session is closed.

Caution: *Do not delete the telnet session filters while tracing is active. This causes traffic to increase exponentially, rendering the connection unusable.*

interface tracedump [<interface>]

The **interface tracedump** command outputs a set of commands that, if executed, resets the interface's trace options to their current settings. If the <interface> is not specified, **tracedump** outputs commands for all interfaces that are not clear.

You can put the current trace settings in a file by piping **tracedump** output to the **output** or **tee** command as shown in the following example.

```
NB:Top> tracedump en1 |output trace.cnf
```

To later restore the trace settings saved in the TRACE.CNF file, enter a **source trace.cnf** command.

```
NB:Top> source trace.cnf
Clearing trace flags, as options would produce no output
en1 disabled
en1 disabled compressed uncompressed raw input output status ip
```

Notice in the above example that the trace options are set, but trace is not enabled until you enter a **trace** <interface> **resume** command.

interface tracelimit [<interface>] [<limit>]

The **interface tracelimit** command sets the maximum number of packets that are traced for a specified interface. Each time a packet is traced, the interface's trace limit is decremented by 1. When the interface's limit reaches 0, tracing for that interface is disabled.

Setting an interface's trace limit to 0 allows unlimited tracing. Therefore, if you resume tracing an interface that was disabled by the trace limit function, tracing for that interface is unlimited, unless you reset the interface's trace limit to a non-zero value.

Entering a **tracelimit** <interface> command without specifying the <limit> value displays the interface's current trace limit setting. Entering **tracelimit** with no parameters displays the trace limits for all interfaces that have a non-zero trace limit value.

IP Commands

The **ip** command takes you to the Configure IP command level. To view a list of the commands available at this level, type a question mark (?) at the Configure IP command level.

```
NB:Top> ip
NB:Top>Configure>IP> ?
Available commands:
address          broadcast        domain>          filter>
icmp>            list             netmask          rip>
route>           rtimer           tcp>             top>
ttl              vj               up>              ?
```

Commands at this level configure and manage the IP (Internet Protocol) facilities. Commands followed by an angle bracket (>) in the above list take you to a subcommand level. For example, entering **route** at this level takes you to the Configure IP Route command level.

ip address [<ip-addr>[/<bits>]] [<interface>|<group-name>|<line> [<port>]]

The **ip address** command sets the global IP address assigned to the NetBlazer, sets the IP address assigned to a specified interface or line, or assigns an IP address to a characteristics group.

Entering this command without any parameters displays the global IP address of the NetBlazer.

```
NB:Top>Configure>IP> address
Global IP address is 0.0.0.0
```

The NetBlazer is shipped with the global IP address set to 0.0.0.0. The next example illustrates how to set the NetBlazer's global IP address.

```
NB:Top>Configure>IP> address 143.191.11.5
Set IP address(es) of 3 interface(s)
Set IP address(es) of 10 lines(s)
```

The global IP address is automatically assigned to each interface and line.

Note: *Only one LAN interface (Ethernet or Token Ring) may use the NetBlazer's global address; each additional LAN interface must be assigned its own unique IP address.*

The next example illustrates how to assign an IP address to the en0 interface.

```
NB:Top>Configure>IP> address 143.191.11.5/24 en0
adding new route 143.191.11.5/24 en0
Set IP address of interface 'en0'
```

The /24 in the previous example tells the NetBlazer that the first 24 bits are the network part of the IP address. It uses this information to automatically create a route through the specified interface and an ARP broadcast entry for the LAN segment connected directly to the NetBlazer's en0 port.

```
NB:Top>Configure>IP> route listlocal
Dest            Len Interface    Gateway          Metric  P Timer  Use
143.191.20.1    32  tbupdate                      1       P 0      0
127.0.0.1       32  loopback                      1       P 0      0
143.191.11.0    24  en0                           1         0      0
NB:Top>Configure>IP> arp list
received 2 badtype 0 bogus addr 0 reqst in 2 replies 0 reqst out 4
IP addr         Type  Interface   Time Q Addr
143.191.11.255  ether any         0      ff:ff:ff:ff:ff:ff
```

An **ip address** <ip-addr> <line> command sets the IP address of a specified line to an address that is different than the NetBlazer's global IP address.

```
NB:Top>Configure>IP> address 143.191.11.253 line10
```

This allows you to telnet to the line from another host that has a route to the line's host or network address.

```
napa > telnet 143.191.63.10
Trying 143.191.63.10 ...
Connected to 143.191.63.10.
Escape character is '^]'.
Connected to modem pool line "line10"
```

When the connection is established, you can issue modem commands, such as **ate1v1** or **at&f**, just as if you were directly connected to the line through a talk session.

You also can assign an IP address to a characteristics group. This capability allows you to establish a modem pool, and use telnet to automatically select a modem that has the proper characteristics (as specified by the characteristics group) for use.

```
NB:Top>Configure>IP> address 143.191.63.253 v32dialout
```

The above example associates the address 143.191.63.253 with any modem that has the characteristics specified by the v32dialout characteristics group.

When the following telnet command is executed, the NetBlazer selects a modem that matches the characteristics specified by the v32dialout group for use by the telnet session.

```
napa > telnet 143.191.63.253
Trying 143.191.63.253 ...
Connected to 143.191.63.253.
Escape character is '^]'.
Connected to modem pool line "line10"
```

Issuing an **ip address** <ip-addr> <line> <port> command changes the TCP port number that is assigned to a line for raw TCP connections, as described in the *NetBlazer Configuration Guide*.

ip broadcast [<ip-addr> [<interface>]]

The **ip broadcast** command sets the interface's broadcast address. If you do not specify an interface, this command sets the global IP broadcast address. Changing the global IP broadcast address also changes the broadcast address for all interfaces that use the global IP address.

Entering an **ip broadcast** command without any parameters displays the current global IP address.

If no **ip broadcast** command is issued, the global IP broadcast address and all interface broadcast addresses default to 255.255.255.255. This is the standard broadcast address that should be used in most cases.

ip domain

The **ip domain** command takes you to the Configure IP Domain command level. To view a list of the commands available at this level, type a question mark (?) at the Configure IP Domain command level.

```
NB:Top> ip domain
NB:Top>Configure>IP>Domain> ?
Available commands:
add             cache           delete          flush
list            lookup          suffix          trace
top>            up>             ?
```

These commands control the domain name service client. If your network includes one or more domain name servers, you can configure the NetBlazer to query the server(s) for the IP address of a specified host.

ip domain add <server-id>

The **ip domain add** command adds a domain name server to the list of name servers that the NetBlazer queries when it needs to match a name to an IP address.

The <server-id> parameter is the IP address of the name server that is to be added to the list. If you have another name server running, you can enter the domain name of the new name server to be added to the list.

ip domain cache

The **ip domain cache** command displays a list of names and IP addresses that have been cached locally. When a name is resolved by a remote name server, its associated address is cached locally. This reduces the load on the network, because the NetBlazer searches the local cache before it queries the remote name server.

ip domain delete <server-id>

The **ip domain delete** command removes a domain name server from the list of name servers that the NetBlazer queries to match a name to an IP address.

The <server-id> parameter is the IP address or domain name of the name server to remove from the list.

ip domain flush

The **ip domain flush** command removes all entries from the domain cache.

ip domain list

The **ip domain list** command displays the IP addresses and status of the name servers that the NetBlazer queries for name resolution.

ip domain lookup <hostname>

The **ip domain lookup** command looks up the specified host's IP address and adds it to the domain cache.

ip domain suffix [<suffix-string>]

The **ip domain suffix** command specifies the <suffix-string> that is added to a name that does not include a suffix; i.e., any name that does not contain a dot (period). For example, Telebit's domain suffix, telebit.com, is set by issuing the following command to the NetBlazer:

```
NB:Top>Configure>IP>Domain> suffix telebit.com
```

If this command is entered with no parameter, the current domain suffix is displayed. To remove a suffix, use a null string in the **suffix** command.

```
NB:Top>Configure>IP>Domain> suffix ""
```

ip domain trace [on|off]

The **ip domain trace** command allows you to monitor domain name server requests and replies.

Issuing a **domain trace** command without specifying on or off shows the current setting of the domain trace function. The default setting of the domain trace function is off.

A **domain trace on** command turns on the domain trace function. When domain tracing is turned on, the NetBlazer displays a trace of queries to the domain name server(s).

Trace output is sent to only one serial port or telnet session – the last one used to issue a **domain trace** command.

A **domain trace off** command turns off the domain trace function.

ip filter

The **ip filter** command takes you to the Configure IP Filter command level. To view a list of the commands available at this level, type a question mark (?) at the Configure IP Filter command level.

```
NB:Top> filter
NB:Top>Configure>IP>Filter> ?
Available commands:
add             delete          deny            flush
list            logdeny         lookup          notify
permit          top>            up>             ?
```

Commands at this level maintain a list of IP packet filters that restrict IP packet routing through a particular interface. Before attempting to set up IP packet filtering, first make sure that the NetBlazer routing table is set up correctly.

ip filter add <address>[/<bits>] [<address>[/<bits>]] <interface> [<proto> [port]] [source|destination] [in|out] permit|deny

The **ip filter add** command adds one or more IP packet permit or deny filters for a designated interface to the list of IP packet filters. The **ip filter permit** or **ip filter deny** commands also allow you to add permit or deny filters to the list of filters.

The following display shows several examples of the **ip filter add**, **permit**, and **deny** commands.

```
NB:Top> ip filter
NB:Top>Configure>IP>Filter> add 143.191/16 syn0 permit
NB:Top>Configure>IP>Filter> deny 143.191.13/24 syn0 tcp <1024 !=25 dest in
NB:Top>Configure>IP>Filter> permit 128.10.2.1 143.191.13/24 syn0 in
NB:Top>Configure>IP>Filter> list
Source              Destination         Interface Protocol I/O  Permit/Deny   Use
128.10.2.1/32       143.191.13.0/24     syn0               In   Permit          0
                    143.191.13.0/24     syn0       TCP     In   Deny            0
                                          port rules: !=25 <1024
143.191.0.0/16                          syn0               In   Permit          0
                    143.191.0.0/16      syn0               In   Permit          0
143.191.0.0/16                          syn0               Out  Permit          0
                    143.191.0.0/16      syn0               Out  Permit          0
Default                                 syn0                    Deny
notify on
```

Notice that input and/or output filtering for a specified interface can be performed on a source address, a destination address, or a source-destination address pair. You can further restrict the filter to apply only to a specified protocol and port number or range.

When packet filtering is turned on for a particular interface, a default filter for that interface is automatically created that is the complement of the first filter. If the first filter is a permit filter, the default filter is set to be a deny filter. On the other hand, if the first filter is a deny filter, the default filter is set to be a permit filter.

The permit|deny parameter determines if the packet filter is a permit or deny filter. A permit filter allows IP packets that match the filter criteria to pass through the designated interface; whereas a deny filter discards IP packets that match the filter's criteria and sends an ICMP Destination Unreachable message back to the source. Packets that do not meet the criteria of a filter are passed on to the next filter in the list.

The <interface> parameter specifies the name of the interface to which the filter applies. Packet filtering is done on a per interface basis. If no filters exist for a particular interface, filtering for that interface is considered off, and all packets routed through that interface will be permitted. When you add the first filter for an interface to the filter list, packet filtering is turned on for that interface, and all packets routed into or out of the interface are checked for permission to continue.

Note: *If an interface is assigned filters in only one direction (i.e., all inbound or all outbound filters), the default filter for that interface only applies to packets traveling in the same direction. Packets going in the opposite direction are automatically permitted.*

The <address>[/<bits>] parameters specify the address or address pair that are trapped by the filter. Entering the keyword **default** in place of the address parameters allows you to change the permission setting of the default filter that is automatically created for a designated interface.

When a pair of addresses are entered, the first address identifies the source network (or host) address, and the second address specifies the destination address. If only one address is entered, the source|destination parameter determines whether this address is to be treated as a source or destination address.

Note: *When creating a filter for a source and destination address pair, do not enter the source|destination parameter.*

If the source|destination parameter is not specified for a single address, the NetBlazer creates both source and destination filters for that address.

The optional in|out parameter determines if the specified filter will filter IP packets entering or leaving the NetBlazer through the designated interface. If this parameter is not specified, filters for both inbound and outbound packets are created.

The optional <proto> and <port> parameters specify the criteria for matching an IP packet's protocol and destination port number. If the <port> parameter is entered, the <proto> parameter must also be specified.

The <proto> parameter is any of the following keywords: **tcp**, **udp**, or **icmp**.

Note: *Do not use the <port> parameter with the icmp protocol.*

The <port> parameter can be just the port number or a range of port numbers defined in Table IP-1.

Table IP-1. Port Number Range Definitions

Syntax	Description
$=P_n$	Any port number equal to P_n. Port numbers entered without a modifier have an implied equal- to characteristic.
$\!P_n$	Excludes any port number equal to P_n. If you enter a exclude command (!) from the keyboard with the **history** function turned on, you must escape the ! by preceding it with a \ (e.g., **ip filter add 143.191.10.0/24 en0 \!25 permit**). If you enter the command from a file or with history turned off, you do not need the escape character (\) before the exclude character (!).
$!=P_n$	Excludes any port number equal to P_n. This is same as the previous command except the exclude character (!) does not need to be escaped when it is followed by an equal sign (=).
$>P_n$	Any port number greater than P_n.
$>=P_n$	Any port number greater than or equal to P_n.
$<P_n$	Any port number less than P_n.
$<=P_n$	Any port number less than or equal to P_n.
$P_1<P_2$	Any port number greater than P_1 and less than P_2.
P_1-P_2	Any port number equal to or greater than P_1 and less than or equal to P_2. This is the same as $P_1<= <=P_2$.
$P_1<= <=P_2$	Any port number equal to or greater than P_1 and less than or equal to P_2. A space is required between the beginning and ending range parameters.
$P_1< <=P_2$	Any port number greater than P_1 and less than or equal to P_2. The space is required.
$P_1<= <P_2$	Any port number equal to or greater than P_1 and less than P_2. The space is required.

To specify port numbers 20, 21, 23, and 25, you could enter any of the following port number range specifications.

```
20<= <=25 !=22 !=24
19<26 !=22 !=24
19< <=25 !=22 !=24
20<= <26 !=22 !=24
```

Refer to RFC 1010 for a list of well-known port numbers used by the TCP and UDP protocols. Services available on UNIX systems can often be found in the /etc/services file.

ip filter delete <address>[/<bits>] [<address>[/<bits>]] <interface> [<proto>] [source|destination] [in|out]

The **ip filter delete** command removes one or more filters from the filter list depending on how specific you are when entering the **ip filter delete** command parameters. More specific commands remove fewer filters from the list. For example, an **ip filter delete** <address/[<bits>] <interface> command removes all filters matching the specified address and interface; whereas an **ip filter delete** <address/[<bits>] <interface> **source in** command only deletes the inbound filter with a matching source address and interface.

To delete one or more filters that have an assigned protocol, you must include the protocol parameter in the **filter delete** command.

The default filter for a particular interface may not be deleted until all the other filters for that interface are deleted. When the last filter is deleted, the default filter is also removed and filtering for the interface is turned off.

ip filter deny <address>[/<bits>] [<address>[/<bits>]] <interface> [<proto> [port]] [source|destination] [in|out]

The **ip filter deny** command adds one or more deny filters to the list of filters. Refer to the **ip filter add** command for a description of the command parameters.

ip filter flush [<interface>]

The **ip filter flush** command removes all filters assigned to a specified interface from the filter list. If you enter this command without the <interface> parameter, you are asked if you want to remove all filters.

ip filter list <interface>

The **ip filter list** command displays a list of packet filters currently in use for the specified interface. Entering this command without the optional <interface> parameter displays a list of filters for all interfaces.

If the IP packet filter list for an interface is empty, all packets routed through that interface are allowed to pass through unimpeded. If one or more filters have been applied to an interface, only the IP packets that meet the criteria of a permit filter are allowed to continue through the interface.

The list of filters for an interface are arranged in the following order:

- **Inbound filters** – apply to packets entering the NetBlazer through the designated interface.
- **Outbound filters** – apply to packets leaving the NetBlazer through the interface.
- **Default filter** – applies to any packet that was not trapped by one of the inbound or outbound filters. Note that if an In or Out is listed in the I/O field of the default filter, the filter only applies to packets traveling in that direction. Packets going in the opposite direction are automatically permitted.

Filters within the above grouping are listed in the order of precedence. For example, a filter at the top of the inbound filter list takes precedence over filters further down in the list of inbound filters for a particular interface. The same is true for outbound filters.

ip filter logdeny <interface> [on|off|<count>]

The **ip filter logdeny** command turns on or off logging of packets that are trapped by an interface's deny filters. Issuing this command without specifying on or off shows the current setting of the denied packet logging for the interface.

A **filter logdeny** <interface> **on** command starts sending messages to syslog every time a packet is filtered by the interface's deny filters.

To prevent overwhelming syslog with denied packet messages, you can specify a <count> that tells the NetBlazer to send syslog a denied packet message for every Nth denied packet. For example, **filter logdeny** <interface> **10** sends syslog a denied packet message for every 10th packet denied by the interface's deny filters.

A **filter logdeny** <interface> **off** command turns off denied packet logging for the specified interface.

ip filter lookup <address>[/<bits>] <address>[/<bits>] <interface> [<proto> [port]]

The **ip filter lookup** command determines whether or not an IP packet is allowed to pass through the NetBlazer based on the packet's source address, destination address, protocol, and destination port number.

The two <address>[/<bits>] parameters specify the source and destination addresses of the packet in that order.

The <interface> parameter identifies the input interface of the packet.

The optional <proto> and <port> parameters indicate the protocol (i.e., **tcp**, **udp**, or **icmp**) and destination port number of the packet.

To determine if a packet will be allowed to pass through the NetBlazer, it performs the following lookup operations:

- Filter lookup for permission to route the packet through the input interface.
- Routing table lookup to determine if a route exists for the destination address.
- Filter lookup for permission to send the packet out through the output interface.

All three conditions must be satisfied to allow the packet to continue through the NetBlazer toward its destination.

ip filter notify [on|off]

The **ip filter notify** command controls whether or not a Destination Unreachable message is sent when a packet is denied by the NetBlazer filters.

Entering this command without specifying on or off displays its current setting. The NetBlazer is shipped with notify set to send a Destination Unreachable message when a packet is denied. Entering a **ip filter notify off** command configures the NetBlazer to silently discard packets that are denied.

ip filter permit <address>[/<bits>] [<address>[/<bits>]] <interface> [<proto> [port]] [source|destination] [in|out]

The **ip filter permit** command adds one or more permit filters to the filter list. Refer to the **ip filter add** command for a description of the command parameters.

The **ip icmp list** command produces a list of ICMP statistics that is maintained as part of the Management Information Base (MIB) used by ICMP.

```
NB:Top> icmp list
( 1)icmpInMsgs                104     (14)icmpOutMsgs                17
( 2)icmpInErrors                0     (15)icmpOutErrors               0
( 3)icmpInDestUnreachs          0     (16)icmpOutDestUnreachs         0
( 4)icmpInTimeExcds             0     (17)icmpOutTimeExcds            0
( 5)icmpInParmProbs             0     (18)icmpOutParmProbs            0
( 6)icmpInSrcQuenchs            2     (19)icmpOutSrcQuenchs           2
( 7)icmpInRedirects             0     (20)icmpOutRedirects            0
( 8)icmpInEchos                85     (21)icmpOutEchos               17
( 9)icmpInEchoReps             17     (22)icmpOutEchoReps            85
(10)icmpInTimestamps            0     (23)icmpOutTimestamps           0
(11)icmpInTimestampReps         0     (24)icmpOutTimestampReps        0
(12)icmpInAddrMasks             0     (25)icmpOutAddrMasks            0
(13)icmpInAddrMaskReps          0     (26)icmpOutAddrMaskReps         0
```

The left-hand column lists all the ICMP messages received. The right hand column lists all the ICMP message sent. Table IP-2 provides a list of the field names and definitions.

Table IP-2. ICMP List Field Definitions

Field Name	Description
icmpInMsgs	The total number of ICMP messages received by the NetBlazer since it was started.
icmpInErrors	Number of error messages received.
icmpInDestUnreachs	Number of destination unreachable messages received.
icmpInTimeExcds	Number of time exceeded messages received.
icmpInParmProbs	Number of parameter problem messages received.
icmpInSrcQuench	Number of source quenches received.
icmpInRedirects	Number of redirect messages received.
icmpInEchos	Number of echo requests (pings) received.
icmpInEchoReps	Number of echo replies (responses to pings) received.
icmpInTimestamps	Number of time stamp requests received.
icmpInTimestampReps	Number of time stamp replies received in response to our time stamp requests.
icmpInAddrMasks	Number of address masks received.
icmpInAddrMaskReps	Number of address mask replies received in response to our address mask requests.

Table IP-2. ICMP List Field Definitions (Continued)

Field Name	Description
icmpOutMsgs	The total number of ICMP messages sent by the NetBlazer since it was started.
icmpOutErrors	Number of error messages sent.
icmpOutDestUnreachs	Number of destination unreachable messages sent.
icmpOutTimeExcds	Number of time exceeded messages sent.
icmpOutParmProbs	Number of parameter problem messages sent.
icmpOutSrcQuench	Number of source quenches sent.
icmpOutRedirects	Number of redirect messages sent.
icmpOutEchos	Number of echo requests (pings) sent.
icmpOutEchoReps	Number of echo replies (responses to pings) sent.
icmpOutTimestamps	Number of time stamp requests sent.
icmpOutTimestampReps	Number of time stamp replies sent in response to time stamp requests.
icmpOutAddrMasks	Number of address masks sent.
icmpOutAddrMaskReps	Number of address mask replies sent in response to address mask requests.

ip list

The **ip list** command produces a list of IP statistics that is maintained as part of the Management Information Base (MIB) used by SNMP.

```
jade:Top> ip list
( 1)ipForwarding                1     ( 2)ipDefaultTTL                64
( 3)ipInReceives            20622     ( 4)ipInHdrErrors                0
( 5)ipInAddrErrors              0     ( 6)ipForwDatagrams          18079
( 7)ipInUnknownProtos           0     ( 8)ipInDiscards                 0
( 9)ipInDelivers            45379     (10)ipOutRequests            33389
(11)ipOutDiscards               0     (12)ipOutNoRoutes                0
(13)ipReasmTimeout             30     (14)ipReasmReqds                88
(15)ipReasmOKs                 44     (16)ipReasmFails                 0
(17)ipFragOKs                 532     (18)ipFragFails                369
(19)ipFragCreates            3708     (20)ipRoutingDiscards        12651
Cache Hits: 14699, Packets Routed: 17238,        ipReasmPendings: 0
Hit Rate: 85 %
```

Table IP-3 provides a list of the field names and definitions.

Table IP-3. IP List Field Definitions

Field Name	Description
ipForwarding	forwarding(1), not-forwarding(2). The NetBlazer only does forwarding(1).
ipDefaultTTL	The default value inserted into the TTL (Time-To-Live) field in IP datagrams. TTL is the maximum number of hops that a packet can traverse before being tossed.
ipInReceives	Number of IP datagrams received from all interfaces.
ipInHdrErrors	Number of IP datagrams discarded due to bad header formats (bad checksum, bad version number, TTL exceeded, etc).
ipInAddrErrors	Number of IP datagrams discarded due to addressing errors.
ipForwDatagrams	Number of IP datagrams received and forwarded.
ipInUnknownProtos	Number of received IP datagrams received with unsupported protocols.
ipInDiscards	Number of inbound IP datagrams discarded due to lack of resources.
ipInDelivers	Number of IP datagrams successfully delivered.
ipOutRequests	Number of IP datagrams queued to be sent (does not include ipForwDatagrams).
ipOutDiscards	Number of outbound IP datagrams discarded due to lack of resources.
ipOutNoRoutes	Number of outbound IP datagrams discarded because of no route to destination.
ipReasmTimeout	Maximum number of seconds the NetBlazer attempts to reassemble a fragmented IP datagram.
ipReasmReqds	Number of IP fragments that required reassembly.
ipReasmOKs	Number of IP datagrams successfully reassembled.
ipReasmFails	Number of failures of IP reassembly (not necessarily either a datagram or a fragment count, just number of errors).
ipFragOKs	Number of successfully fragmented IP datagrams.
ipFragFails	Number of unsuccessfully fragmented IP datagrams.
ipFragCreates	Number of IP fragments generated at this NetBlazer.
ipRoutingDiscards	Routes discarded due to lack of resources.

ip netmask [<ip-addr> [<interface>]]

The **ip netmask** command sets the interface's netmask address. The netmask contains a 1 in every bit position that is part of the network address.

If you do not specify an interface, this command sets the global IP netmask address. Changing the global IP netmask also changes the netmask for all interfaces that use the global IP address.

Entering an **ip netmask** command without any parameters displays the current global netmask address.

If no **ip netmask** command is issued, the global IP netmask address and all interface netmask addresses default to the following standard netmasks: 255.0.0.0 for class A addresses, 255.255.0.0 for class B addresses, or 255.255.255.0 for class C addresses.

Note: *Changing an interface's IP address also changes the interface's netmask. Therefore, the **ip netmask** command must be issued after the corresponding **ip address** command.*

ip rip

The **ip rip** command takes you to the Configure IP RIP command level. To view a list of the commands available at this level, type a question mark (?) at the Configure IP RIP command level.

```
NB:Top> ip rip
NB:Top>Configure>IP>RIP> ?
Available commands:
accept          add             delete          list
netbits         merge           refuse          request
timeout         top>            up>             ?
```

Commands at this level configure and manage RIP (Routing Information Protocol). The NetBlazer is shipped with the RIP server turned on. This allows the NetBlazer to accept RIP advertisements from other IP routers on the network. If you do not want the NetBlazer to accept routing information from other routers, you can issue a **server stop rip** command to turn off the RIP server. If you want the NetBlazer to refuse RIP advertisements from a specific IP router, issue a **ip rip refuse** <IP-address> command.

To configure the NetBlazer to send RIP advertisements to other routers on the network, you must add one or more entries to the list of destinations to which the NetBlazer sends routing information.

ip rip accept <gateway>

The **ip rip accept** command configures the NetBlazer to start accepting RIP advertisements from a gateway that was previously refused. This reverses the effects of a **ip rip refuse** command.

ip rip add <destination> <interval> [<split-horizon>]

The **ip rip add** command adds a destination IP address as the recipient for RIP information.

The <destination> parameter specifies the destination IP address for RIP advertisements sent from the NetBlazer. This is usually a broadcast IP address but may be a specific IP address for another router. Make sure that destination address is on the same LAN segment as the NetBlazer.

The <interval> parameter specifies the number of seconds between RIP transmissions to the destination address.

The optional [<split-horizon>] parameter enables or disables split horizon, a technique that improves the performance of RIP when a router disappears. The default value for this parameter is "yes" to enable split horizon. The "no" option disables split horizon.

ip rip delete <destination>

The **ip rip delete** command removes an IP address from the RIP destination list. The <destination> parameter specifies which IP address to remove.

ip rip list

The **ip rip list** command displays the list of destinations to which the NetBlazer sends routing information.

ip rip netbits [<value>]

The **ip rip netbits** command sets the number of network bits for all RIP route advertisements received by the NetBlazer.

Entering this command without specifying a <value> displays the current setting. The default setting is 0, which causes the NetBlazer to set a RIP acquired route's bit mask based on the route's address class.

Setting netbits to a nonzero value applies the specified bit mask setting to all routes acquired from RIP. For example, entering an **ip rip netbits 28** sets the bit mask for routes acquired from RIP to 28.

Note: *If after the bit mask is applied the host address is nonzero, the address is treated as a host address (i.e., the bit mask is set to 32).*

ip rip merge [on | off]

The **ip rip merge** command turns RIP merging on or off. This potentially reduces the size of the routing table by merging the routing table entries as much as possible.

The NetBlazer is shipped with RIP merging turned off. When you turn on RIP merging, you should flush the routing table for the merge to take effect.

ip rip refuse <gateway>

The **ip rip refuse** command tells the NetBlazer to ignore routing information sent from the specified gateway IP address. The <gateway> parameter is the IP address of the gateway from which routing information will be refused.

ip rip request <gateway> [<port-number>]

The **ip rip request** command immediately requests routing information from a specified gateway rather than waiting for the next scheduled RIP update.

ip rip timeout <interface> <seconds>

The **ip rip timeout** command sets the amount of time that routes acquired through the specified interface are kept in the routing table. If the route is not updated via RIP within the time allowed, the route is discarded. The default timeout value is 240 seconds.

Caution: *Make sure that an interface's RIP timeout value is greater than the RIP updating interval; otherwise the routes acquired through the interface will not be available from the time the routes time out until the next RIP update is received.*

ip route

The **ip route** command takes you to the Configure IP Route command level. To view a list of the commands available at this level, type a question mark (?) at the Configure IP Route command level.

```
NB:Top> ip route
NB:Top>Configure>IP>Route> ?
Available commands:
add             addprivate      balance         change
delete          flush           list            listbalance
listlocal       lookup          metric          private
public          top>            up>             ?
```

Commands at this level maintain the NetBlazer routing table.

When the RIP server is running, RIP periodically updates the RIP acquired routes in the routing table.

The **ip route add** or **addprivate** commands add static routes to the routing table. The **ip address** or **dynamic ipdial** commands also add static routes to the routing table. Static routes designated as private are not advertised by RIP to other routers on the network.

Temporary routes are automatically added to the routing table when the NetBlazer establishes a connection through a dynamic interface that has auto-routes set by the **dynamic auto-route** or **auto-arp** commands. These routes are removed from the routing table when the dynamic interface connection is dropped.

Balanced routes are routes to the same remote destination through multiple dynamic and/or dedicated interfaces. The **ip route balance** command specifies which interfaces to include in a balanced route.

ip route add <dest-addr>[/<bits>] <interface> [<gateway>] [<metric>]

The **ip route add** command adds a static route to the routing table.

The <dest-addr> parameter is the destination network (or host) IP address, and the optional /<bits> parameter specifies the number of bits assigned to the network and subnetwork part of the IP address. This parameter must be preceded by a slash (/) with no spaces between the destination IP address and the number of bits.

The <interface> parameter specifies the name of the interface through which the packets are routed.

The optional <gateway> and <metric> parameters specify the IP address of the gateway that the packets should be sent to for further routing and the number of hops required to reach the destination network. To set a route's metric without specifying a gate, enter a null ("") or a 0 in the <gateway> parameter's position.

Note: *The NetBlazer allows you to add a route with the same destination address as an existing route. If these routes are not balanced, the NetBlazer uses the first entry in the table that meets the destination address routing criteria.*

The following commands add three new static routes to the routing table.

```
NB:Top>Configure>IP>Route> add 143.191.167/24 en0
NB:Top>Configure>IP>Route> add 143.191.13/24 en0 143.191.167.254 2
NB:Top>Configure>IP>Route> addprivate default en0 143.191.167.1 2
```

The first command adds a route to a network that is connected directly to the NetBlazer through the en0 interface.

The second command adds a route to a nonadjacent network, which is also accessed through the en0 interface. However, IP packets destined for this network are sent to the gateway with an IP address of 143.191.167.254. The last parameter in this command indicates that the destination network is 2 hops away.

The last command adds a private default route used to route packets that do not have a specific route defined.

ip route addprivate <dest-addr>[/<bits>] <interface> [<gateway>] [<metric>]

The **ip route addprivate** command is equivalent to the **ip route add** command, except routes added to the routing table through this command are not advertised to other routers by RIP.

ip route balance <dest-addr>[/<bits>] <interface> <interface> [<interface> ...]

The **ip route balance** command balances traffic bound for a specified destination over a group of interfaces.

To use load balancing in both directions, you must issue a **ip route balance** command on both sides of the link.

If you want to delay bringing up all of the lines until the traffic through the first connection warrants it, issue an **interface mdelay** command for each interface in a balanced route.

Refer to the *NetBlazer Configuration Guide* for instructions on how to set up a multiline load balanced link between two NetBlazers.

If you want to turn balancing off, you must delete the balanced routes. If you still want the routes in the table, but unbalanced, reenter the routes with the **ip route add** command.

ip route change <dest-addr>[/<bits>] <interface> [<gateway>] [<metric>]

The **ip route change** command changes the interface, gateway, and metric of an existing static route specified by the <dest-addr>[/<bits>] parameter. Routes acquired through RIP cannot be changed.

The <interface> parameter specifies the name of the interface through which the packets are routed. If the NetBlazer does not find a route that matches the <dest-addr>[/<bits>] and <interface> parameters, it changes the name of the first route in the table that matches the <dest-addr>[/<bits>] parameter.

The optional <gateway> and <metric> parameters specify the IP address of the gateway that the packets should be sent to for further routing and the number of hops required to reach the destination network.

The following example illustrates how to change the routing for a specified destination.

```
NB:Top>Configure>IP>Route> list |grep 143.191.63.0
143.191.63.0    24  syn0                              1          0       0
NB:Top>Configure>IP>Route> change 143.191.63/24 en0 143.191.65.254 3
NB:Top>Configure>IP>Route> !l
list |grep 143.191.63.0
143.191.63.0    24  en0           143.191.65.254   3          0       0
```

The above **ip route change** command changes the route used to send packets to the 143.191.63/24 subnetwork from sending the packets through the syn0 interface to send the packets to the 143.191.65.254 gateway through the en0 interface. Notice that this example also illustrates how to use the **grep** and **history** features to display the route for a specific destination address.

ip route delete <dest-addr>[/<bits>] [<interface>] [<gateway>]

The **ip route delete** command removes a specified route from the routing table. The <dest-addr>/[<bits>] and optional <interface> and <gateway> parameters identifies which route to remove.

If you do not specify an <interface> and/or <gateway>, the NetBlazer deletes the first route in the table that matches the given parameters.

ip route flush

The **ip route flush** command removes the RIP entries from the routing table. This command is useful immediately after restructuring a network so that the NetBlazer does not need to wait for the routing table entries to expire (timeout) before learning new routes.

ip route list

The **ip route list** command displays the current routing table, as illustrated in the following example:

```
NB:Top>Configure>IP>Route> list
Dest            Len Interface    Gateway          Metric  P Timer  Use
127.0.0.1       32  loopback                      1       P 0      0
143.191.20.1    32  tbupdate                      1       P 0      0
143.191.65.0    24  en0                           1         0      1
143.191.64.0    24  en0          143.191.65.254   2       R 220    0
default         0   en0          143.191.65.254   2       R 220    55
```

Dest – Indicates the IP address of the destination. The default destination routes any IP packet that does not fall within a specific destination range.

Len – Indicates how many bits represent the network and subnetwork part of the IP address.

Interface – Identifies the interface through which the packets are routed.

Gateway – Identifies any router or gateway that a packet should be sent to for further routing.

Metric – Indicates the number of hops required to reach the destination or network.

P – Indicates if a route is designated as private (P), has been acquired through RIP (R), or has been automatically (A) added to the table by a dynamic interface connection.

Timer – Indicates the number of seconds before this route is discarded. A nonzero value indicates that the route was generated by RIP and will expire when the timer reaches 0. A value of 0 indicates that this is a static route and should remain in the table forever.

Use – Indicates the number of IP packets that have used this route since the route was created.

ip route listbalance

The **ip route listbalance** command displays a list of balanced routes that were created with the **ip route balance** command.

ip route listlocal

The **ip route listlocal** command displays a list of static routes that were not acquired through RIP.

ip route lookup <dest-addr>

The **ip route lookup** command checks for a valid route to a destination address. If a valid route exists, the NetBlazer returns the routing table entry used to route packets to that destination. If no route exists, the NetBlazer returns a message indicating that the destination is unreachable.

The <dest-address> parameter specifies the IP address of the destination network. If the NetBlazer is using the domain name system, you can enter a domain name in place of the <dest-address> parameter.

If you look up a destination that is included in a group of balanced routes, the NetBlazer shows the best available route for that instance.

ip route metric <dest-addr>[/<bits>] [<interface>] [<gateway>] <metric>

The **ip route metric** command changes the metric (hop count) of a static route specified by the <dest-addr>[/<bits>] and optional <interface> and <gateway> parameters.

If you do not specify an <interface> and/or <gateway>, the NetBlazer changes the metric of the first route in the table that matches the given parameters.

ip route private <dest-addr>[/<bits>] <interface> [<gateway>]

The **ip route private** command changes a public route (i.e. a route that is advertised by RIP to other routers on the network) to a private route. The <dest-addr>[/<bits>], <interface> and optional <gateway> parameters identify which route to make private.

ip route public <dest-addr>[/<bits>] <interface> [<gateway>]

The **ip route private** command changes a private route (i.e., a route that is not advertised by RIP to other routers on the network) to a public route. The <dest-addr>[/<bits>], <interface> and optional <gateway> parameters identify which route to make public.

ip rtimer [<reassemble-time>]

The **ip rtimer** command sets or displays the maximum time in seconds that IP will wait for all the fragments to reassemble the original datagram.

If this command is entered without the <reassemble-time> parameter, the current value is displayed.

ip tcp

The **ip tcp** command takes you to the Configure IP TCP command level. To view a list of the commands available at this level, type a question mark (?) at the Configure IP TCP command level.

```
NB:Top> ip tcp
NB:Top>Configure>IP>TCP> ?
Available commands:
irtt            list            mss             reset
top>            up>             window          ?
```

These commands configure and manage TCP (Transmission Control Protocol).

ip tcp irtt [<int-rnd-trip-time>]

The **ip tcp irtt** command sets or displays the initial round trip time in milliseconds used by TCP when opening a new TCP session with a remote host. Entering this command without a parameter shows the current IRTT value.

ip tcp list [<tcb>]

The **ip tcp list** command produces a list of TCP statistics, including an entry for each TCP connection. The connection display is preceded by the address of the TCP control block (tcb).

```
NB:Top>Configure>IP>TCP> list
( 1)tcpRtoAlgorithm                4      ( 2)tcpRtoMin                      5
( 3)tcpRtoMax             2147483647      ( 4)tcpMaxConn            4294967295
( 5)tcpActiveOpens                 7      ( 6)tcpPassiveOpens                4
( 7)tcpAttemptFails                0      ( 8)tcpEstabResets                 0
( 9)tcpCurrEstab                   0      (10)tcpInSegs                   2308
(11)tcpOutSegs                  2311      (12)tcpRetransSegs                57
(13)tcpInErrs                      0      (14)tcpOutRsts                     0
#    &TCB Rcv-Q Snd-Q  Local socket          Remote socket          State
02 1a3cd0     0     0  0.0.0.0:21           0.0.0.0:0              Listen (S)
03 1a48c4     0     0  0.0.0.0:79           0.0.0.0:0              Listen (S)
04 1a0e20     0     0  0.0.0.0:23           0.0.0.0:0              Listen (S)
```

If a particular tcb is specified, all the information about that particular TCP connection is displayed. Here is a sample display of the information about the tcb at address e9ab0:

```
NB:Top>Configure>IP>TCP> list 1a3cd0
Local: 0.0.0.0:21 Remote: 0.0.0.0:0 State: Listen
      Init seq    Unack     Next Resent CWind Thrsh  Wind  MSS Queue       Total
Send:        0        0        0      0   512 65535     1  512     0           0
Recv:        0                 0      0               2048          0           0
Timer stopped SRTT 5000 ms Mean dev 0 ms
```

ip tcp mss [<max-seg-size>]

The **ip tcp mss** command sets or displays the maximum segment size for a TCP session. If this command is entered without a parameter, the current MSS value is displayed. MSS is the number of bytes that a NetBlazer will tell a remote host that it may send as a single segment.

ip tcp reset <tcb>

The **ip tcp reset** command resets the TCP session specified by the <tcb> parameter. The <tcb> parameter is derived from the **tcp list** command.

Note: *The **session kill** command is a better choice for killing a session where the remote host is no longer responding.*

ip tcp window [<max-window-size>]

The **ip tcp window** command sets or displays the maximum window size that will be presented to a remote TCP during a session. The actual window size varies as a function of the TCP protocol, but it will never exceed the value set by this command. If this command is entered without a parameter, the current maximum window size is displayed.

ip ttl [<time-to-live>]

The **ip ttl** command sets the value placed in the time-to-live field of an IP datagram originating from within the NetBlazer.

If this command is entered without the <time-to-live> parameter, the current value is displayed.

ip vj <interface> [on|off|auto]

The **ip vj** command controls whether or not Van Jacobson TCP/IP header compression is used for a specified SLIP or PPP interface.

Entering an **ip vj** <interface> command without specifying the setting displays the current setting for the specified interface.

Using VJ Header Compression on a SLIP Interface

Table IP-4 shows the possible VJ header compression setting for a SLIP connection. We recommend that you leave one end of the link set to its default value of auto and use the other end to turn VJ compression on or off.

By default, VJ header compression is set to auto, which means that the interface starts with header compression turned off, but it is automatically turned on when a compressed packet arrives from the remote site. Once header compression is turned on, it is not turned off until the connection is dropped.

Using VJ Header Compression on a PPP Interface

VJ header compression for a PPP connection is negotiated between the local and remote end while establishing a connection. Once header compression is negotiated on or off, it remains in effect for the duration of the connection.

Table IP-5 shows the possible VJ header compression settings and results for a PPP connection.

Table IP-4. VJ Header Compression for SLIP Connections

Local	Remote	Result and Comments
auto	auto	VJ header compression is off. However, it can be turned on by issuing a **vj on** command from either the local or remote end.
	off	VJ header compression is disabled. It can be turned on by issuing a **vj on** command from the remote end.
	on	VJ header compression is enabled. When header compression is turned on, it cannot be turned off for the current connection.
off	auto	VJ header compression is disabled. It can be turned on by issuing a **vj on** command from the local end.
	off	VJ header compression is disabled.
	on	This is an illegal setting, which causes one end of the link to attempt to use header compression while the opposite end is not using header compression. To avoid this condition, set at least one end of the link to auto.
on	auto	VJ header compression is enabled. When header compression is turned on, it cannot be turned off for the current connection.
	off	This is an illegal setting, which causes one end of the link to attempt to use header compression while the opposite end is not using header compression. To avoid this condition, set at least one end of the link to auto.
	on	VJ header compression is enabled. When header compression is turned on, it cannot be turned off for the current connection.

Table IP-5. VJ Header Compression for PPP Connections

Local	Remote	Result and Comments
auto	auto	VJ header compression is negotiated to **on** when a connection is established.
	off	VJ header compression is negotiated to **off** when a connection is established.
	on	VJ header compression is negotiated to **on** when a connection is established.
off	auto	VJ header compression is negotiated to **off** when a connection is established.
	off	VJ header compression is negotiated to **off** when a connection is established.
	on	VJ header compression is negotiated to **off** when a connection is established.

Table IP-5. VJ Header Compression for PPP Connections (Continued)

Local	Remote	Result and Comments
on	auto	VJ header compression is negotiated to **on** when a connection is established.
	off	VJ header compression is negotiated to **off** when a connection is established.
	on	VJ header compression is negotiated to **on** when a connection is established.

IPX Commands

The **ipx** command takes you to the Configure IPX command level. To view a list of the commands available at this level, type a question mark (**?**) at the Configure IPX command level prompt.

```
NB:Top> ipx
NB:Top>Configure>IPX> ?
Available commands:
acs>            add             address         bang
delete          frames          ifconfig        internal-net
iptunnel>       keepalive>      list            log
netbios         netstat         ping            rip>
route>          sap>            top>            up>
wan>            ?
```

Commands at this level are used to configure and manage the IPX (Internet Packet eXchange) facilities. Commands followed by an angle bracket (>) in the above list take you to a subcommand level. For example, if you enter **route** at this level, the NetBlazer takes you to the Configure IPX Route command level.

ipx acs

The **ipx acs** command takes you to the Configure IPX ACS command level. To view a list of the commands available at this level, type a question mark (**?**) at the Configure IPX ACS command level prompt.

```
NB:Top>Configure>IPX> acs
NB:Top>Configure>IPX>ACS> ?
Available commands:
audit           deregister      list            lock
start           stop            top>            up>
?
```

Use the commands at this level to configure and manage the NetBlazer Asynchronous Communications Server (ACS). NetBlazer ACS is an option you can purchase separately.

ipx acs audit [on|off]

The **ipx acs audit** command turns on or off the NetBlazer ACS audit log facility. When the audit log is on, the NetBlazer sends audit log messages to the NetBlazer syslog utility. The default is off.

ipx acs deregister <user>

The **ipx acs deregister** command removes a workstation from the list of workstations registered at the NetBlazer. Normally, you do not use this command. If a workstation is removed from the network, you can use this command to remove the workstation from the list of registered workstations.

ipx acs list

The **ipx acs list** command displays a list of connection information and the list of registered workstations. The connection information is similar to that displayed by the NetBlazer ACS's management console utility, PDVIEW.EXE. This utility is supplied on a separate diskette with other NetBlazer ACS programs.

ipx acs lock <line-name> [on|off]

The **ipx acs lock** command tells the NetBlazer to lock the configuration for a line. Once the line configuration is locked, a terminal emulator cannot change line speed or flow control. However, you can change a line configuration by using the NetBlazer line commands.

The <line-name> parameter specifies which asynchronous line's configuration to lock. Examples are line01 and line13.

ipx acs start

The **ipx acs start** command starts NetBlazer ACS.

ipx acs stop

The **ipx acs stop** command stops NetBlazer ACS.

ipx add <interface>|all

The **ipx add** command sets up an existing NetBlazer interface for use by IPX.

The <interface> parameter specifies the name of the interface to which packets are routed.

The all parameter lets you set all currently available interfaces for use by IPX. Use the **configure interface list** command to see a list of currently available interfaces.

Note: *SLIP interfaces are not available to IPX.*

The first of the following sample commands adds an Ethernet interface. The second command is an example of adding a dynamic interface.

```
NB:Top>Configure>IPX> add en0
NB:Top>Configure>IPX> add Ruby
```

ipx address [<ipx-addr> [<interface>]]

The **ipx address** command sets the global IPX address assigned to the NetBlazer or sets the IPX address assigned to a specified interface or line. Work with your network administrator in assigning IPX addresses.

An IPX address has three parts, and each part is identified by a maximum number of hexadecimal characters, as shown in Table IPX-1:

Table IPX-1. Parts of an IPX Address

Part	Maximum Number of Hexadecimal Characters
Network	8
Host	12
Socket	4

For the NetBlazer and most interfaces, enter only the network number (for example, 331) or both the network and host numbers separated by a period (for example, 300.10). Let the NetBlazer or NetWare assign the socket (or port) number unless you are assigning the address to a process that gets a well-known socket number.Some standard socket numbers are shown in Table IPX-2; numbers are hexadecimal:

Table IPX-2. IPX Socket Numbers

Socket Number	Meaning
0001h	Xerox Network Standard (XNS) RIP
0002h	Echo protocol packet
0003h	Error handler packet

Table IPX-2. IPX Socket Numbers

Socket Number	Meaning
0020h-003Fh	Experimental
0001h-0BB8h	Registered with Xerox
0BB9h-	Dynamically assigned
0451h	NetWare file service packet (NCP)
0452h	NetWare service advertising packet
0453h	NetWare RIP
0455h	NetBios packet
0456h	diagnostic packet
4001h	NetWare watchdog packet
4003h	NetWare shell packet
4004h-7FFFh	Dynamically assigned sockets
8000h-	Socket numbers assigned by Novell

The optional <interface> parameter specifies the name of the interface to which packets are sent and through which the packets are routed.

If you enter this command without any parameters, the NetBlazer displays its global IPX address.

```
NB:Top>Configure>IPX> address
Global IPX address is 0
```

The NetBlazer is shipped with the global IPX address set to 0. The next example illustrates how to set the NetBlazer's global IPX address:

```
NB:Top>Configure>IPX> address 300
Global IPX address is 300
```

The global IPX address is automatically assigned to each interface.

Each Ethernet and Token Ring interface must have a unique IPX address. Use an **ipx address** <ipx-addr> <interface> command to set the IPX address of an interface to an address that is different from the NetBlazer's global IPX address. This example shows how to set up the IPX address for a specific Ethernet interface:

```
NB:Top>Configure>IPX> address 17 en1
Set IPX address of interface 'en1'
```

Make sure the IPX network number you give each Ethernet and Token Ring interface matches the network's IPX address for that interface.

ipx bang <ipx-addr> [<length> [<count> [<interval> [<outstanding>]]]]

The **ipx bang** command sends IPX echo requests to a specified remote NetBlazer. This command is similar to an **ipx ping** command except **ipx bang** prints only a single character (!) instead of a line of statistics for each response. Lost or out-of-sequence packets are shown as a dot (.). IPX messages are shown as an X in the **bang** display. Refer to the **ipx ping** command for an explanation of the parameters used by both commands.

ipx delete <interface>|all

The **ipx delete** command removes either one IPX interface or all IPX interfaces, depending on which parameter, <interface> or all, you enter. This command also deletes all IPX routes associated with the interface(s).

ipx frames <interface> [ethernet_802.3|ethernet_2|token-ring|token-ring _snap]

The **ipx frames** command specifies framing for Ethernet and Token Ring interfaces. The default for Ethernet is Ethernet 802.3. The default for Token Ring is Token Ring Snap.

If you use Ethernet II packets, use the **frames** <interface> **ethernet_2** command to specify the correct frames.

If you have Token Ring and you have not changed the file server to Token Ring Snap, use the **frames** <interface> **token-ring** command to specify the correct frames.

ipx ifconfig <interface> [<address>] [up|down]

The **ipx ifconfig** command provides another way to set the IPX address of an interface. You cannot set the global IPX address with the **ipx ifconfig** command, so use the **ipx address** command instead of the **ipx ifconfig** command.

The <interface> parameter specifies the name of the interface to which packets are sent and through which the packets are routed.

The optional <address> parameter is the interface's address.

The optional down parameter disables an interface. The optional up parameter enables an interface after it was brought down by an **ipx ifconfig down** command. You can bring down an interface only if you have brought it up or added it.

ipx internal-net <ipx-network-number>

The **ipx internal-net** command sets the internal network number for the NetBlazer router. You need this for NetBlazer ACS.

Work with your network administrator in assigning an internal network number. An internal network number can be up to eight hexadecimal characters. See Table IPX-1 for the parts of an IPX address. Each internal network number must be different from any other network number or internal network number. NetWare 3.11 assigns a node (or host) number of 1 to the internal network number.

ipx iptunnel

The **ipx iptunnel** command takes you to the Configure IPX IpTunnel command level. To view a list of the commands available at this level, type a question mark (**?**) at the Configure IPX IpTunnel command level prompt.

```
NB:Top> ipx iptunnel
NB:Top>Configure>IPX>IpTunnel> ?
Available commands:
add             delete          flush           list
route           top>            up>             ?
```

Commands at this level are used to configure and manage IP (Internet Protocol) tunneling. The command **iptunnel** provides a means of encapsulating IPX datagrams within IP packets so IPX traffic can travel across an IP internetwork. (An IPX datagram is carried in the data portion of an IP packet.) This allows IPX networks to be connected across internetworks that carry only IP traffic.

ipx iptunnel add <ip-addr>

The **ipx iptunnel add** command adds a peer IP address. To use an IP internetwork, the NetBlazer keeps a list of the IP addresses of other IPX routers on the IP internetwork. This is the peer list and each address on the list is a peer IP address. Make this list with the **ipx iptunnel add** command. The peer list is the list of hosts to which the NetBlazer sends IPX broadcast packets such as Routing Information Protocol (RIP) and Service Advertising Protocol (SAP) packets.

ipx iptunnel delete <ip-addr>

The **ipx iptunnel delete** command deletes a peer IP address from the peer list.

ipx iptunnel flush

The **ipx iptunnel flush** command removes all peer IP addresses from the peer list.

ipx iptunnel list

The **ipx iptunnel list** command displays the list of all peer IP addresses.

ipx iptunnel route <ipx-dest> <ip-gateway> [<metric>]

The **ipx iptunnel route** command adds an IPX iptunnel route through an IP gateway to a destination IPX network.

The <ipx-dest> parameter identifies the IPX destination, that is, the destination on a remote IPX network. An example is the IPX address of a file server on the destination IPX network.

The <ip-gateway> parameter specifies the IP address of the gateway to which the IPX packets should be sent, for example, the IP address of a NetBlazer on the destination IPX network.

The optional <metric> parameter specifies the number of hops required to reach the destination IPX network. The default value for <metric> is 1.

ipx keepalive

The **ipx keepalive** command takes you to the Configure IPX Keepalive command level. To view a list of the commands available at this level, type a question mark (**?**) at the Configure IPX Keepalive command level prompt.

```
NB:Top> ipx keepalive
NB:Top>Configure>IPX>Keepalive> ?
Available commands:
list            notify          start           stop
timeout         top>            up>             ?
```

Use commands at this level to manage IPX keep-alive messages and responses.

The NetWare watchdog timer checks idle workstations and keeps their connections with the file server "alive" as long as the workstation is running.

People using NetWare log in to servers but often continue to do most of their work on their local drives. They look to the remote server for information infrequently. When the NetBlazer makes a connection, a server sends keep-alive messages to the workstation to make sure it is still active. It would be wasteful to keep a dynamic interface active just to transfer these keep-alive messages and responses. This can be prevented with the **ipx keepalive** command.

With the **ipx keepalive** command, the remote NetBlazer (the one at the server's end) pretends to be the remote workstation and responds to keep-alive packets. The local NetBlazer (the one at the workstation end) monitors the workstation and sends keep-alive packets to it as long as it is active. If the workstation changes to the server's drive, the dial-up link is reestablished. If the workstation is down, the local NetBlazer notifies the server that the workstation is down.

You can set the idle time with the NetBlazer command **interface dynamic timeout** (make sure the idle timer is not less than 360 seconds). The IPX keep-alive packets do not update the idle timer. If a line has no other traffic than IPX keep-alive packets, RIP packets, and SAP packets, the line is considered to be idle. If a line is idle longer than the maximum idle time, the dial link is taken down.

ipx keepalive list

The **ipx keepalive list** command displays the list of keep-alive entries. Keep-alive entries are the file servers and workstations with which the NetBlazer is maintaining a logical connection with keep-alive packets.

ipx keepalive notify [on|off]

The **ipx keepalive notify** command is turned on when the **IPX keepalive** command is started. With this command on, the NetBlazer notifies the remote NetBlazer on the dial-up link of the status of keep-alive entries.

The command also displays the list of keep-alive entries.

You can use the command to turn off keep-alive notification. The NetBlazer is shipped with keep-alive notification turned on.

ipx keepalive start

The **ipx keepalive start** command starts IPX keep-alive packets. The NetBlazer is shipped with keep-alive started.

ipx keepalive stop

The **ipx keepalive stop** command stops IPX keep-alive packets.

ipx keepalive timeout [<max-time(minutes)>]

The **ipx keepalive timeout** command sets the maximum time, in minutes, for sending IPX keep-alive packets. The NetBlazer rounds up the entered value to the nearest 15 minutes. When the time is reached, the NetBlazer stops sending IPX keep-alive packets. The default specifies no limit.

ipx list

The **ipx list** command displays status information about the IPX interfaces currently used by the NetBlazer router.

ipx log [<facility> [<level>]]

The **ipx log** command is a debugging utility that sends error and debugging messages to the syslog utility.

The **ipx log** command without parameters displays a table showing activity for ACS, Net, RIP, and SAP. Net covers IPX and PPP interfaces.

The optional <facility> parameter is one of these: **acs**, **net, rip**, or **sap**.

The optional <level> parameter has the range 0 through 7 and specifies the level of tracing. The higher the number, the more trace information you receive. The default level is 3 (this means only errors go to the syslog).

ipx netbios <interface> [on|off]

The **ipx netbios** command controls the forwarding of netbios broadcasts. Netbios clients use netbios broadcasts to locate netbios servers. IPX routers forward these broadcasts across interfaces.

By default, forwarding of netbios broadcasts is on for all interfaces. Use the **ipx netbios** command to disable the forwarding of netbios broadcasts on an interface.

The <interface> parameter specifies the name of an interface over which netbios broadcast forwarding is enabled or disabled.

Netbios broadcasts are not forwarded over a dynamic link if the link is down. Netbios broadcasts are "spoofed", that is, the dynamic link that has only netbios broadcasts triggers the idle timer and brings the link down.

ipx netstat [-Aalimrs] | [-p ipx|spx]

The **ipx netstat** command displays a table of running statistics for packet traffic on configured network interfaces. The table shows cumulative statistics for packets transferred, errors, collisions, the network addresses for the interface, and the maximum transmission units (mtu).

Note: *The tables can change while netstat is examining them, creating incorrect or partial displays.*

The parameters work as described in Table IPX-3.

Table IPX-3. Netstat Parameters

Parameter	Function
-A	Shows address of protocol control blocks related to sockets.
-a	Shows the status of all sockets.
-l	Shows information for local routes (use with -r parameter).
-i	Shows the condition of interfaces.
-m	Shows statistics for an IPX network's private buffer pool.
-p ipx|spx	Selects protocol.
-r	Shows routing table (with -s, shows routing statistics instead).
-s	Shows protocol statistics (with -r, shows routing statistics instead).

ipx ping <ipx-addr> [<length> [<count> [<interval> [<outstanding>]]]]

The **ipx ping** command sends IPX echo requests to a specified remote NetBlazer router. A NetBlazer pings only another NetBlazer. If the first attempt to ping is not successful, enter **^C** (control-C) and ping the NetBlazer a second time.

The <ipx-addr> parameter is the IPX address of the remote NetBlazer router.

The optional <length> parameter is used to pad the request to a larger size specified in bytes. This parameter must be greater than 32 bytes (the minimum length allowed) to take effect. However, a value less than 32 can be used as placeholder in the command. The default length is 64 bytes.

The optional <count> parameter specifies the number of IPX requests to send to the remote host. If you do not enter this parameter or if you enter 0 as a placeholder, the NetBlazer continues issuing requests to the host until you stop the ping session by pressing ctrl-C.

The optional <interval> parameter specifies in milliseconds the time to wait between sending IPX echo requests. If you can set this value to 0, the NetBlazer sends pings with 10 milliseconds separation (approximately 100 pings per second). The default interval value is 0 for 10 milliseconds.

The optional <outstanding> parameter specifies the number of echo requests (pings) that may be transmitted before the NetBlazer stops and waits for their responses. The default specifies no limit.

The ability to set the ping interval to 0 (10 milliseconds between pings) and set the number of outstanding pings is a useful tool for placing a significant load on a network without overloading the network.

ipx rip

The **ipx rip** command takes you to the Configure IPX RIP command level. (The **ipx rip** command is not the same as the **ip rip** command.)

To view a list of the commands available at this level, type a question mark (**?**) at the Configure IPX RIP command level prompt.

```
NB:Top> ipx rip
NB:Top>Configure>IPX>RIP> ?
Available commands:
add             delete          list            listen
stat            top>            up>             ?
```

Commands at this level are used to configure and manage the IPX RIP. IPX RIP learns about routes to other indirect networks by receiving IPX routing table entries from other routers. The NetBlazer can participate in this process by transmitting its own IPX routing tables using the **ipx rip add** command.

ipx rip add <interface> [<interval (minutes)>]

The **ipx rip add** command starts RIP broadcasts on the network or interface. The NetBlazer is shipped with IPX RIP broadcasting turned off.

The <interface> parameter specifies the name of an interface over which IPX RIP broadcasts are sent periodically.

The optional <interval> parameter specifies the number of minutes between IPX routing table transmissions to the interface. The default is one minute. This is the recommended interval for a connection.

ipx rip delete <interface>

The **ipx rip delete** command stops broadcasting IPX RIP packets to an interface.

The <interface> parameter specifies which IPX interface should be removed from the IPX RIP interface list.

This command stops IPX RIP broadcasts to the interface.

ipx rip list

The **ipx rip list** command displays the list of interfaces to which the NetBlazer is sending routing information as well as the interfaces to which the NetBlazer is listening for routing information and requests.

ipx rip listen <interface>|all on|off

The **ipx rip listen** command tells the NetBlazer to put one or all current IPX interfaces in the listen state. You can also use the command to turn off the listen state for one or all interfaces. The NetBlazer is shipped with IPX RIP listening turned on.

ipx rip stat

The **ipx rip stat** command displays a table of IPX RIP statistics. The table shows the number of queries, responses, replies, broadcasts, send errors, and packets with incorrect length. It also shows a timer. Use this command only when debugging IPX RIP.

ipx route

The **ipx route** command takes you to the Configure IPX Route command level. To view a list of the commands available at this level, type a question mark (**?**) at the Configure IPX Route command level prompt.

```
NB:Top> ipx route
NB:Top>Configure>IPX>Route> ?
Available commands:
add             addprivate      bang            delete
flush           list            listlocal       ping
stat            top>            up>             ?
```

IPX route commands are used to maintain the NetBlazer IPX routing table.

Use the **ipx route add** or **addprivate** commands to add static routes to the IPX routing table. Static routes that are designated as private are not advertised by RIP to other routers on the network. (Also, the **ipx address** command adds static routes for only Ethernet and Token Ring networks to the IPX routing table.)

Dynamic routes are added to the IPX routing table through RIP.

ipx route add <dest-addr> <interface> [<gateway> [<metric>]]

The **ipx route add** command is used to add a static route to the IPX routing table.

The <dest-addr> parameter is the destination network (or host) IPX address.

The <interface> parameter specifies the name of the interface through which the packets are routed.

The optional <gateway> and <metric> parameters specify the IPX address of the gateway that the packets should be sent to for further routing and the number of hops required to reach the destination network.

To specify a route with a metric but no gateway, enter a gateway of **0** or " ".

Caution: *If you add a route with the same destination address as an existing route, the system generates an error message. Delete the old route before adding the new route.*

The following commands add three new static routes to the IPX routing table.

```
NB:Top>Configure>IPX>Route> add 410 Pearl
NB:Top>Configure>IPX>Route> add 350 Ruby "" 2
```

ipx route addprivate <dest-addr> <interface> [<gateway> [<metric>]]

The **ipx route addprivate** command is equivalent to the **ipx route add** command with one exception: Routes added to the IPX routing table through this command are not advertised to other routers by RIP.

ipx route bang <ipx-addr> [<length> [<count> [<interval> [<outstanding>]]]]

The **ipx route bang** command sends IPX echo requests to a specified remote NetBlazer. This command is similar to an **ipx route ping** or **ipx ping** command except **ipx route bang** prints only a single character (!) instead of a line of statistics for each response. Lost or out-of-sequence packets are shown as a dot (.). IPX messages are shown as an X in the **bang** display. Refer to the **ipx ping** command for an explanation of the parameters used by both commands.

ipx route delete <dest-addr> [<gateway>]

The **ipx route delete** command removes a specified route from the IPX routing table. The <dest-addr> parameter and the optional <gateway> parameter identify which route to remove.

ipx route flush

The **ipx route flush** command removes the IPX routing table entries that were created by RIP. This leaves only the static IPX routing table entries (those entered by hand). This command is useful immediately after restructuring a network so that the NetBlazer will not have to wait for the IPX routing table entries to expire (timeout) before learning new routes.

The **ipx route list** command displays the current IPX routing table:

```
NB:Top>Configure>IPX>Route> list
Routing tables
Destination     Gateway            Flags   Refs     Use   Hops Ticks
Interface
321.*           330.E831           UG        0        0     1   37  topaz
331.*           330.E831           UG        0        0     1   37  topaz
334.*           334.80AD034C46     U         1        0     0    1  en0
```

Destination – Indicates the IPX address of the destination.

Gateway – Identifies any router or gateway that a packet should be sent to for further routing.

Flags – Indicates if a route is designated as shown in Table IPX-4.

Table IPX-4. Route Flags

Flag	Meaning
A	Auto route
B	Backup route
D	Dynamic
G	Route to a gateway
H	Route to a host
P	Private
U	Route is up

Refs – Indicates whether or not the route is in use now.

Use – Indicates the number of IPX packets that have used this route since the route was created.

Hops – Indicates the number of hops required to reach the destination or network.

Ticks - Indicates the number of ticks it takes for a packet to reach its destination. A tick is one-eighteenth (1/18) of a second.

Interface – Indicates the interface or device through which packets will be routed.

ipx route listlocal

The **ipx route listlocal** command displays a list of static routes that were not acquired through IPX RIP.

ipx route ping <ipx-addr> [<length> [<count> [<interval> [<outstanding>]]]]

The **ipx route ping** command sends IPX echo requests to a specified remote NetBlazer. This command is similar to an **ipx ping** command. Refer to the **ipx ping** command for an explanation of the parameters used by the **ipx route ping** command.

ipx route stat

The **ipx route stat** command displays a list of IPX routing statistics. The list shows the number of bad routing redirects, dynamically created routes, new gateways due to redirects, destinations found unreachable, and uses of a wildcard route. Use this command only when debugging an IPX route.

ipx sap

The **ipx sap** command takes you to the Configure IPX SAP command level. To view a list of the commands available at this level, type a question mark (**?**) at the Configure IPX SAP command level prompt.

```
NB:Top> ipx sap
NB:Top>Configure>IPX>SAP> ?
Available commands:
add             delete          list            listen
stat            table>          top>            up>
?
```

Commands at this level are used to configure and manage the IPX SAP. SAP learns about services by receiving service table entries from the servers.

The NetBlazer advertises its services to servers and it records in a service table the routes to the servers. Normally, SAP accepts service information from any server. You may not want the NetBlazer to accept service information from a particular server or servers.

ipx sap add <interface> [<interval (minutes)>]

The **ipx sap add** command starts SAP broadcasts on an interface. The NetBlazer is shipped with IPX SAP broadcasting turned off.

The <interface> parameter specifies the name of the interface.

The optional <interval> parameter specifies the number of minutes between SAP transmissions to the interface. The default is one minute. This is the recommended interval for a connection.

ipx sap delete <interface>

The **ipx sap delete** command stops SAP broadcasts on an interface.

The <interface> parameter specifies the name of the interface.

ipx sap list

The **ipx sap list** command displays a list of the names of interfaces to which the NetBlazer is sending service information. It also lists the names of interfaces to which the NetBlazer is listening for service information.

ipx sap listen <interface>|all on|off

The **ipx sap listen** command starts listening for SAP broadcasts on an interface.

You can also use the command to turn off the listen state for one or all interfaces. The NetBlazer is shipped with SAP listening turned on.

ipx sap stat

The **ipx sap stat** command displays SAP statistics. The statistics show the number of queries, responses, replies, broadcasts, send errors, table size, and packets with incorrect length or bad type. It also shows a timer count. Use this command only when debugging IPX SAP.

ipx sap table

The **ipx sap table** command takes you to the Configure IPX SAP Table command level. To view a list of the commands available at this level, type a question mark (**?**) at the Configure IPX SAP Table command level prompt.

```
NB:Top> ipx sap table
NB:Top>Configure>IPX>SAP>Table> ?
Available commands:
add             delete          list            top>
up>             ?
```

Commands at this level are used to add and delete service table entries.

ipx sap table add [<name> [<type> [<ipx-addr>]]]

The **ipx sap table add** command adds service table entries. The parameters discussed below are optional only in that you do not have to enter them when you enter the **sap table add** command. You can enter the command with no parameters. If you do so, the NetBlazer prompts you for the missing parameters.

The optional parameter <name> is the server name. The NetBlazer converts the name you enter to upper case.

The optional parameter <type> is the server type. Table IPX-5 lists some well known server types; numbers are hexadecimal:

Table IPX-5. Server Types

Type	Meaning
FFFFh	Wild
0000h	Unknown
0003h	Print queue
0004h	File server
0005h	Job server
0006h	Gateway
0007h	Print server
0009h	Archive server
000Bh	Administration
0021h	NAS SNA gateway

Table IPX-5. Server Types

Type	Meaning
0024h	Remote bridge server
0027h	TCP/IP gateway
002Dh	Time synchronization server
0047h	Advertising print server
004Ah	Telebit ACS server
007Ah	TES – NetWare for VMS
0098h	Netware access server
009Eh	Portable NetWare
0133h	NNS domain
8000h	Reserved up to

The optional parameter <ipx-addr> is the server's IPX address in hexadecimal. The three parts of an IPX address are network, node, and socket. The socket number for a file server is 451h, for example. For some IPX socket numbers, see Table IPX-2 on page -3 in this section.

ipx sap table delete [<name> [<type>]]

The **ipx sap table delete** command deletes service table entries.

ipx sap table list

The **ipx sap table list** command displays the list of service table entries.

```
NB:Top>Configure>IPX>SAP>Table> list
Type Server Name                 Hops F Interface  Address
   4 PLUTO                          3 S topaz      00000321 000000000001
0451
```

Type – Indicates server type. See Table IPX-5 for server types.

Server Name – Lists the name you assign to the server.

F - Flag. **S** shows the entry is static; otherwise the column is blank.

Hops – Indicates the number of hops required to reach the server.

Interface – Indicates the interface or device through which the server is reached.

Address – Shows the server's IPX address in hexadecimal. See Table IPX-2 for IPX socket numbers.

ipx wan

The **ipx wan** command takes you to the Configure IPX WAN command level. To view a list of the commands available at this level, type a question mark (**?**) at the Configure IPX WAN command level prompt.

```
NB:Top> ipx wan
NB:Top>Configure>IPX>WAN> ?
Available commands:
add             delete          list            stat
top>            up>             ?
```

Commands at this level are used to enable or disable Novell's IPX Wide Area Network (WAN). IPX WAN can be used in place of Point-to-Point Protocol (PPP).

ipx wan add <interface>

The **ipx wan add** command enables IPX WAN on an interface.

ipx wan delete <interface>

The **ipx wan delete** command disables IPX WAN on an interface.

ipx wan list [<interface>]

The **ipx wan list** command lists WAN interfaces and their states. This command is usually used for debugging.

ipx wan stat

The **ipx wan stat** command displays IPX WAN statistics. The statistics cover the timer, state, retries, length, sequence, errors, type, and identification. It also shows a timer count. Use this command only when debugging IPX WAN.

Line Commands

The **line** command takes you to the Configure Line command level. To view a list of the commands available at this level, type a question mark (?) at the Configure Line command prompt.

```
NB:Top> line
NB:Top>Configure>Line> ?
Available commands:
access          auto            callin          command
characteristics ctstime         dcdtime         delay
destgroup       dsrtime         dtrtime         encode
flowcontrol     idle            list            login
modem           promptdelay     request-call    reset
speed           terminal        test-request    trace
top>            up>             v25bis          ?
```

These commands control the operating characteristics of the NetBlazer's serial ports.

line access <line-name> [packetlall]

The **line access** command configures the NetBlazer to allow only packet mode connections from the outside on a specified line. This prevents callers from using the line as a terminal server connection to the NetBlazer (refer to Figure LIN-1).

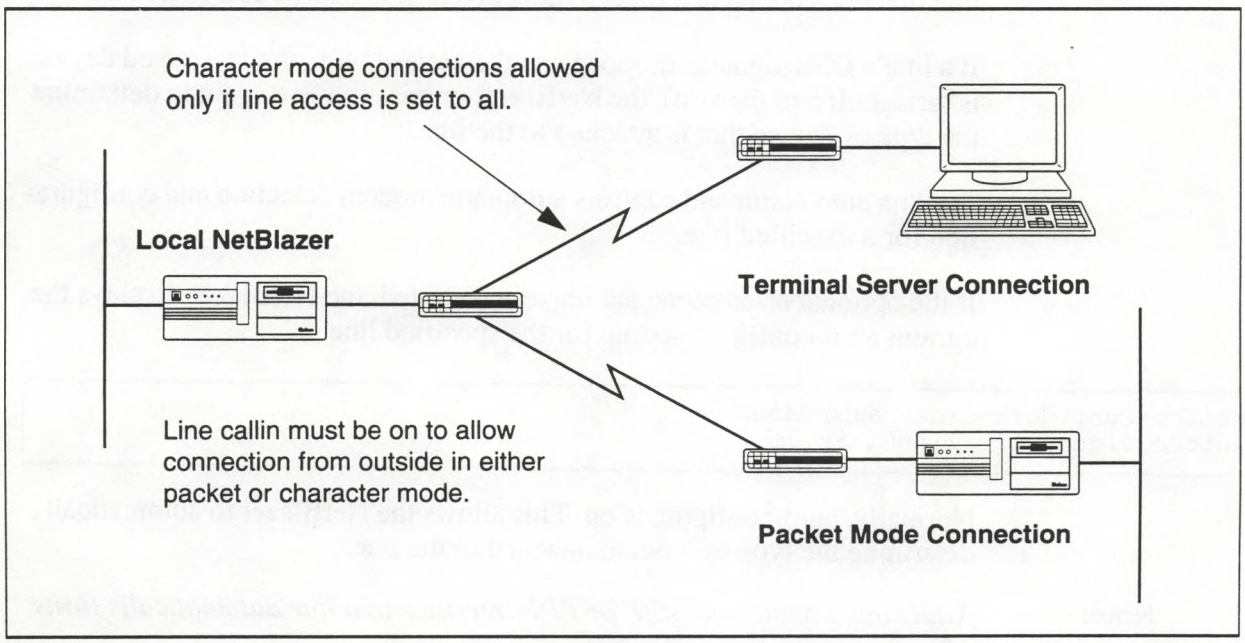

Figure LIN-1. Restricting Line Access

If the optional packet|all parameter is omitted, the current access setting for the specified line is displayed.

```
NB:Top>Configure>Line> access line01
'line01' access is all
```

Normally, line access is set to all, which allows any outside call to establish a connection with the NetBlazer.

Issuing a **line access** <line-name> **packet** command sets the line specified by the <line-name> parameter to only allow packet mode connections for incoming calls.

```
NB:Top>Configure>Line> access line01 packet
'line01' accesscontrol is packet only
```

Issuing a **line access** <line-name> **all** command sets the line to allow all incoming calls.

```
NB:Top>Configure>Line> access line01 all
'line01' accesscontrol is all
```

line auto <line-name> [on|off|config]

When the NetBlazer is turned on or rebooted, it automatically detects the type of modem connected to the line and preconfigures the line and the modem based on information contained in the MODEMCAP.TXT file. Any line that does not have a modem attached to it is treated as a login line.

If a line's DSR signal is dropped and then raised (e.g., the connected device is turned off and then on), the NetBlazer queries the line again to determine the type of device that is attached to the line.

The **line auto** command controls automatic modem detection and configuration for a specified line.

If the optional on|off|config parameter is omitted, the NetBlazer displays the current auto-configure setting for the specified line.

```
NB:Top>Configure>Line> auto line01
Autoconfigure of  line01 is on
```

Normally, auto-configure is on. This allows the NetBlazer to automatically determine the type of modem attached to the line.

Note: *Attaching a dedicated SLIP or PPP interface to a line automatically turns off auto-configure for that line.*

Issuing a **line auto** <line-name> **off** command turns off auto-configure for the specified line.

```
NB:Top>Configure>Line> auto line01 off
Autoconfigure of line01 is off
```

Issuing a **line auto** <line-name> **config** command turns off automatic modem type detection but allows auto-configuration for the modem type defined by the **line modem** command.

```
NB:Top>Configure>Line> modem line01 t3000
NB:Top>Configure>Line> auto line01 config
Autoconfigure of line01 config
```

Issuing a **line auto** <line-name> **on** command returns the specified line to its default state, which allows automatic modem detection and configuration.

```
NB:Top>Configure>Line> auto line01 on
Autoconfigure of line01 is on
```

line callin <line-name> [on|off]

The **line callin** command designates the specified line as dialout-only line. When a line's callin setting is set to off, the NetBlazer sends an ATS0=0 to the modem. This tells the modem to stop answering calls. The NetBlazer also disables line's login prompt.

Refer to the **line access** command description for additional information on controlling incoming call connections to the NetBlazer.

If the optional on|off parameter is omitted, the NetBlazer displays the current callin setting for the specified line.

```
NB:Top>Configure>Line> callin line01
Callin on line 'line01' is on
```

Normally, line callin is set to on, which allows remote users to use the line for inbound and outbound calls.

Issuing a **line callin** <line-name> **off** command designates the specified line for outbound calls only.

```
NB:Top>Configure>Line> callin line01 off
Callin on line 'line01' is off
```

Issuing a **line callin** <line-name> **on** command returns the specified line to its default state, which allows both inbound and outbound calls.

```
NB:Top>Configure>Line> callin line01 on
Callin on line 'line01' is on
```

line command <line-name> [<command> [<arguments>]]

The **line command** command allows you to assign a NetBlazer command string to a line. Entering a **line command** command with just a <line-name> parameter removes the NetBlazer command string assigned to the line.

When a call is received on the specified line, the line's assigned command is invoked before the login process. For example, assigning a **telnet** <host> command to a line bypasses the NetBlazer's login and automatically passes the caller through to the specified host.

```
NB:Top>Configure>Line> command line01 telnet robin
```

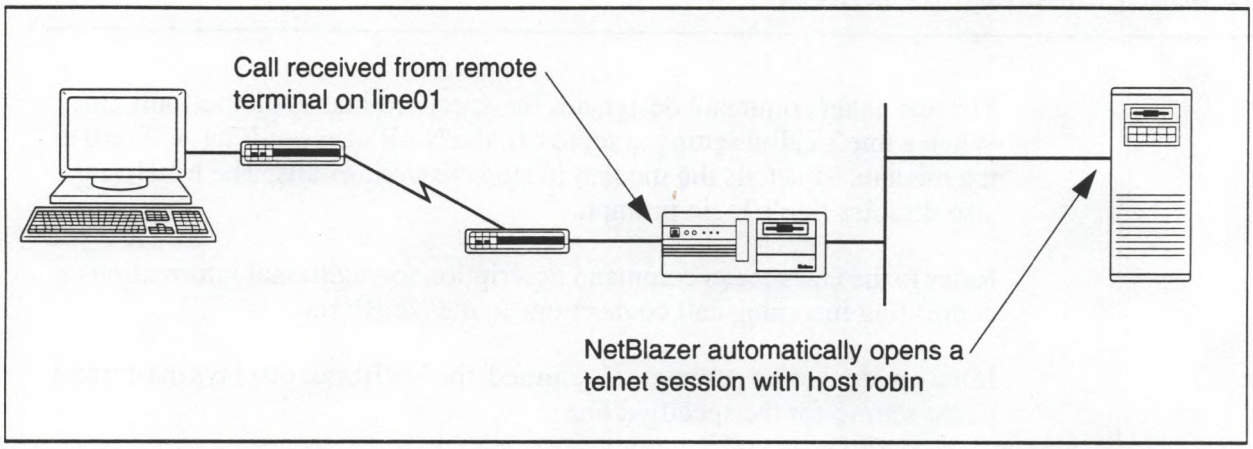

Figure LIN-2. Automatic Telnet

Note: *If you assign a* ***telnet*** *command to a line, as shown in the above example, the line only can be used for establishing a telnet session with the designated host from a remote terminal. Attempting to use the line for a dynamic (packet mode) connection will have unpredictable results.*

If you assign a command to a line that is attached to a terminal with the DCD signal held high, the command is executed each time the line is not in use. For example, if you assign a **telnet robin** command to a line with a terminal attached, the NetBlazer establishes a telnet session between the host named robin and the terminal. This allows a user to log in to robin. When the user logs out of robin, another telnet session is automatically established with robin so that the next user can log in to robin from the terminal.

line characteristics <line-name> [<group-name-of-characteristics>]

The **line characteristics** command assigns a named group of characteristics to a designated line.

The <group-name-of-characteristics> parameter is the name of an existing group that contains the desired characteristics. Refer to the **group** command descriptions for instructions on how to use characteristic groups.

The NetBlazer comes with seven standard dialout characteristic groups: dialout, v32dialout, pepdialout, request, raw, raw_dial, and v25dialout.

The dialout or v32dialout groups should be used for most destinations. If you encounter problems with connections failing because the modems unexpectedly hang up and you have PEP-compatible modems at both sites, try the pepdialout group.

The request characteristics group is automatically assigned to a line that is designated as a callback request line.

The raw and raw_dial characteristics groups configures a line for raw connections. Raw only allows raw connections on the line. Raw_dial allows both raw connections and dialout connections if a modem is attached to the line.

The v25dialout characteristics group configures a synchronous line for use with an ISDN terminal adapter.

line ctstime <line-name> [<seconds.milliseconds>]

The **line ctstime** command sets the minimum time that modem must hold CTS on for the NetBlazer to assume that the modem is ready to accept data.

Issuing this command without specifying a time duration displays the current setting of the CTS lock time.

```
NB:Top>Configure>Line> ctstime line01
'line01' cts lock time is 0.125 seconds
```

The default CTS lock time is 0.125 seconds. To change this time to 0.25 seconds, for example, issue the following **line ctstime** command.

```
NB:Top>Configure>Line> ctstime line01 0.25
'line01' cts lock time is 0.250 seconds
```

line dcdtime <line-name> [<seconds.milliseconds>]

The **line dcdtime** command sets the amount of time that a modem must hold DCD low before the NetBlazer decides that the connection has been dropped.

Entering this command without specifying a time displays the current DCD lock time.

```
NB:Top>Configure>Line> dcdtime line01
'line01' dcd lock time is 0.320 seconds
```

The default DCD lock time for a line is 0.320 seconds. You can extend (or reduce) this time as shown in the next example.

```
NB:Top>Configure>Line> dcdtime line01 0.5
'line01' dcd lock time is 0.500 seconds
```

If you set the DCD lock time to 0, the NetBlazer ignores DCD signal changes.

```
NB:Top>Configure>Line> dcdtime line01 0
'line01' dcd lock time is 0.000 seconds
```

line delay <line-name> [<milliseconds>]

The **line delay** command sets the amount of time that the NetBlazer pauses after the modem has accepted a command and returns a result code.

Entering this command without specifying a time displays the current delay time.

```
NB:Top>Configure>Line> delay line01
'line01' delay time is 0 milliseconds
```

The default delay time is 0 milliseconds. If a Telebit modem is attached to the line, you do not need to change this value. However, if some other modem is attached to the line, you may need to increase the delay time as shown in the following example.

```
NB:Top>Configure>Line> delay line01 50
'line01' delay time is 50 milliseconds
```

destgroup <line-name> [<destination-group>]

The **line destgroup** command prevents users who are not assigned to the specified destination group from logging in on the specified line.

Note: *This restriction only applies to users who are in a destination group that is different than the destination group assigned to the line. It does not prevent users who are not assigned a destination group from logging in on the line.*

Entering this command with just a <line-name> parameter removes the destination group restriction from the line.

line dsrtime <line-name> [<seconds.milliseconds>]

The **line dsrtime** command sets the minimum time that modem must hold DSR on for the NetBlazer to assume that the modem is ready.

Issuing this command without specifying a time duration displays the current setting of the DSR lock time.

```
NB:Top>Configure>Line> dsrtime line01
'line01' dsr lock time is 2.000 seconds
```

The default DSR lock time is 2 seconds. To change this time to 3.5 seconds, for example, issue the following **line dsrtime** command.

```
NB:Top>Configure>Line> dsrtime line01 3.5
'line01' dsr lock time is 3.500 seconds
```

line dtrtime <line-name> [<milliseconds>]

When the NetBlazer needs to use a modem, it first drops DTR and issues a series of AT commands to bring the modem back to a known state.

The **line dtrtime** command sets the time duration that the NetBlazer holds down the DTR signal for a specified line.

Issuing this command without specifying a time duration displays the current amount of time that DTR is held low when resetting a modem attached to the specified line.

```
NB:Top>Configure>Line> dtrtime line01
'line01' dtr off time is 1100 milliseconds
```

The default DTR off time is 1100 milliseconds (1.1 seconds). To extend this time to 2 seconds, for example, issue the following **line dtrtime** command.

```
NB:Top>Configure>Line> dtrtime line01 2000
'line01' dtr off time is 2000 milliseconds
```

To prevent the NetBlazer from dropping DTR to reset a modem on a specified line, set the dtrtime value to 0.

```
NB:Top>Configure>Line> dtrtime line01 0
'line01' dtr off time is 0 milliseconds
```

Note: *Setting a line's dtrtime to 0 prevents the NetBlazer from issuing AT commands to bring the attached modem back to a known state.*

line encode <line-name> [sync|async|nrzi|nrz]

The **line encode** command sets the encoding method used on a synchronous line to either NRZI or NRZ. On a NetBlazer PN, you can also use this command to configure line00 for synchronous or asynchronous mode.

Issuing this command without specifying an encoding method displays the current setting of the line.

```
NB:Top>Configure>Line> encode syn0
line 'syn0' is using NRZ encoding
```

The default encoding method is NRZ. To set a synchronous line's encoding to NRZI, enter a **line encode** <line-name> **nrzi** command.

```
NB:Top>Configure>Line> encode syn0 nrzi
NB:Top>Configure>Line> encode syn0
line 'syn0' is using NRZI encoding
```

To set a synchronous line's encoding method back to its default value of NRZ, enter a **line encode** <line-name> **nrz** command.

```
NB:Top>Configure>Line> encode syn0 nrz
NB:Top>Configure>Line> encode syn0
line 'syn0' is using NRZ encoding
```

The **sync** and **async** parameters only apply to line00 of a NetBlazer PN. This line is initially set to operate in asynchronous mode. To set the line for synchronous operation, issue a **line encode line00 sync** command.

```
NetBlazer PN:Top>Configure>Line> encode line00 sync
NetBlazer PN:Top>Configure>Line> encode line00
line 'line00' is using NRZ encoding
```

To set line00 of a NetBlazer PN back to its default mode, enter a **line encode line00 async** command.

```
PNB:Top>Configure>Line> encode line00 async
```

line flowcontrol <line-name> [none|auto|xon|cts]

The **line flowcontrol** command sets the flow control method used by the NetBlazer to regulate the flow of data between the NetBlazer and the device attached to a specified line.

You can configure a line to use XON/XOFF, CTS/RTS, or no flow control; or you can configure the line to switch automatically between XON/XOFF and CTS/RTS or no flow control.

The default flow control setting for a line is auto flow control. This method uses the following rules to determine when to use XON/XOFF, CTS/RTS, or no flow control:

- Use CTS/RTS flow control when the NetBlazer detects a modem on the line during the auto-configure process.
- Use XON/XOFF flow control when a user is logged in on the line and has a command prompt from the command interpreter.
- Turn off XON/XOFF flow control when the user starts a data communications session, such as telnet or dial, to allow the XON and XOFF characters to pass through the NetBlazer. If the CTS signal is present use CTS/RTS flow control; otherwise leave flow control turned off.

Notes: *To prevent lost or garbled data, both the NetBlazer and the attached device must agree on the flow control method used. The NetBlazer auto-configures Telebit modems to use CTS/RTS flow control. Therefore, lines with Telebit modems attached should use auto flow control or CTS/RTS flow control.*

Any line used for a dynamic interface connection must not be set to use XON/XOFF flow control. Use auto, CTS/RTS, or none.

Issuing **line flowcontrol** command without specifying a flow control method displays the line's current flow control setting.

```
NB:Top>Configure>Line> flowcontrol line01
'line01' flow control is auto
```

The following example shows you how to set a line to use XON/XOFF flow control.

```
NB:Top>Configure>Line> flowcontrol line01 xon
'line01' flow control is xon/xoff
```

The next example shows you how to set a line to use CTS/RTS flow control.

```
NB:Top>Configure>Line> flowcontrol line01 cts
'line01' flow control is cts/rts
```

To set a line to not use flow control, specify **none** as shown below.

```
NB:Top>Configure>Line> flowcontrol line01 none
'line01' flow control is none
```

To configure a line to switch automatically between XON/XOFF and CTS/RTS or no flow control, specify **auto**.

```
NB:Top>Configure>Line> flowcontrol line01 auto
'line01' flow control is auto
```

line idle <line-name> [minutes]

The **line idle** command configures the specified line to reset automatically if it is idle for a set number of minutes.

Entering this command without specifying the number of minutes shows the current line idle setting.

```
NB:Top>Configure>Line> idle line01
line01 idle reset disabled
```

The NetBlazer is shipped with the line idle function disabled for all lines.

The next example shows you how to configure line01 to reset automatically when it is idle for 30 minutes.

```
NB:Top>Configure>Line> idle line01 30
NB:Top>Configure>Line> idle line01
line01 idle reset after 30 minutes
```

Setting line idle to 0 minutes disables the line idle function for the specified line as shown below.

```
NB:Top>Configure>Line> idle line01 0
NB:Top>Configure>Line> idle line01
line01 idle reset disabled
```

line list [<line-name>]

Entering a **line list** command without the optional <line-name> parameter displays a list of all the lines, including the type of modem (if any) attached to the line, the characteristics group name assigned to the line, the current usage of the line, the NetBlazer command (if any) assigned to the line, and the line flags, which includes information about the current status of the line.

```
NB:Top>Configure>Line> list
Line     Modem      Char's      Use        Command    MFlags
line00   <<None>>   <<None>>    login      <<None>>   <CTS,RTS,DTR>
line01   <<None>>   <<None>>    login      <<None>>   <RTS,DTR>
syn0     <<None>>   <<None>>    none       <<None>>   <DTR>
```

To get additional information concerning a specific line, enter the **line list** command with a <line name> parameter. This listing shows the current line parameter settings. For definitions of each parameter, refer to its associated command description.

```
pearl:Top>Configure>Line> list line00
Line     Modem      Char's      Use        Command    MFlags
line00   <<None>>   <<None>>    login      <<None>>   <CTS,RTS,DTR>
Flags:  00180202   <CTS,AUTOFL,RTS,DTR>
DSR: OFF        DCD: OFF        CTS: ON       RTS: ON       DTR: ON
Speed: 9600    Flow control: auto
Login: ON      Callin: ON     Access: all     Autoconfigure: on
Modem command delay: 0 milliseconds
Modem status delay (seconds): DCD=2.000     DSR=3.000     CTS=0.125
DTR off time: 1100 milliseconds
Idle times (minutes):   receive=8332     transmit=8332     auto-reset=0
Trace: OFF        Callback monitor: OFF
IP address: 143.191.63.1
Terminal type: <<None>>        Rows: not set        Cols: not set
Process = line00, Dial Protocol = Console, state = Conn Term
```

The IP address parameter shows the IP address assigned to the line by a **ip address** <ip-addr> <line> [<port>] command. The NetBlazer's global IP address is initially assigned to all lines as a default. If you assign a line a unique IP address, you can telnet to the device attached to the specified line. This can be used to "talk" with a modem over an IP network.

Note: *Configuring a line for a raw connection automatically sets the IP address and port number assigned to a line.*

line login <line-name> [on|off]

The **line login** command determines if a login prompt is sent to the specified line. This command is useful when you want the NetBlazer to accept a call on the line but do not want the NetBlazer to issue a login prompt.

Note: *Creating a dedicated interface automatically turns off* ***login*** *for the line used by that interface.*

Refer to the **line access** and **line callin** command descriptions for additional information on controlling incoming call connections to the NetBlazer.

If the optional on|off parameter is omitted, the current login setting for the specified line is displayed.

```
NB:Top>Configure>Line> login line10
login on line 'line10' is on
```

Normally, line login is set to on, which requires remote users to log in to the NetBlazer when calling in on the specified line.

Issuing a **line login** <line-name> **off** command turns off the login prompt for the specified line.

```
NB:Top>Configure>Line> login line10 off
login on line 'line10' is off
```

Issuing a **line login** <line-name> **on** command returns the specified line to its default state, which requires users to log in to the NetBlazer when calling in on the specified line.

```
NB:Top>Configure>Line> login line10 on
login on line 'line10' is on
```

line modem <line-name> [<modem-name>]

If a line is set to auto-configure, the NetBlazer automatically determines the type of modem attached to the line whenever the DSR signal is raised.

The **line modem** command is used to specify the type of modem attached to a specific line if for some reason it fails to properly identify the modem type.

Before changing the modem type, make sure that the line is set to auto-configure, then try resetting the line by issuing a **line reset** command. If that does not work, use the **line modem** command to set the line's modem type.

```
NB:Top>Configure>Line> modem line01 t2500
NB:Top>Configure>Line> list
Line      Modem       Char's      Use         Command     MFlags
line00    <<None>>    <<None>>    login       <<None>>    <CTS,RTS,DTR>
line01    t2500       <<None>>    login       <<None>>    <RTS,DTR>
syn0      <<None>>    <<None>>    none        <<None>>    <DTR>
```

Note: *The <modem-name> parameter must point to a modem type that is defined in the MODEMCAP.TXT file.*

Issuing a **line modem** command without the <modem-name> parameter sets the line's modem type to <<None>>.

```
NB:Top>Configure>Line> modem line01
NB:Top>Configure>Line> list
Line      Modem       Char's      Use         Command     MFlags
line00    <<None>>    <<None>>    login       <<None>>    <CTS,RTS,DTR>
line01    <<None>>    <<None>>    login       <<None>>    <RTS,DTR>
syn0      <<None>>    <<None>>    none        <<None>>    <DTR>
```

line promptdelay <line-name> [<seconds.milliseconds>]

The **line delay** command sets the amount of time that the NetBlazer pauses before sending a login prompt. This command is useful if you experience a problem with the NetBlazer presenting a login prompt too early. The default behavior is to not delay sending a login prompt.

line request-call <line-name> [on|off]

The **line request-call** command designates the specified line as a callback request monitor line. This is the line that a remote user should call to request a callback.

If the optional on|off parameter is omitted, the current request-call setting for the specified line is displayed.

```
NB:Top>Configure>Line> request-call line01
Request-callbacks on line 'line01' is off
```

Normally, a line's request-call setting is off. To designate a line as a call-back request line, set the line's request-call parameter to on.

```
NB:Top>Configure>Line> request-call line01 on
Request-callbacks on line 'line01' is on
```

To return a callback request line to normal use, set the line request-call parameter to off.

```
NB:Top>Configure>Line> request-call line01 off
Request-callbacks on line 'line01' is off
```

line reset <line-name> [dtroff|dtron]

The **line reset** command resets the specified line. This provides a way to reset a modem that has become unresponsive for any reason. Entering this command causes the NetBlazer to disconnect a call in progress and then reconfigure the line.

```
NB:Top>Configure>Line> reset line01
```

Note: *If the line's auto-configure is turned off, a reset line command resets the line but does not attempt to reconfigure the line.*

If you enter a **line reset** <line-name> **dtron** command, the line's DTR signal stays on throughout the reset process. If you enter a **line reset** <line-name> **dtroff** command, the line's DTR signal stays off. If you do not specify **dtron** or **dtroff**, the NetBlazer first resets the modem (DTR off) and then completes the reset process with DTR on.

line speed <line-name> <speed>

The **line speed** command is used to set the data transfer rate on a specified asynchronous line.

The <speed> parameter may be set to any of the following standard data transfer rates: 300, 1200, 2400, 4800, 9600, 19200, 38400, 57600, or 115200.

Note: *The speed of the connected device must match the line speed.*

The following display shows you how to change the speed of line01 to 19.2K bps.

```
Top>Configure>Line> speed line01 19200
```

If you are using a terminal connected to the specified line, the command prompt displayed after this command is scrambled. Set your terminal to match the speed of the line (e.g., 19.2K bps), and press the return key. The prompt is then displayed correctly.

line terminal <line-name> [<terminal-type>] [-rows <rows>] [-cols <cols>]

The **line terminal** command provides information to a host about the terminal attached to the specified line. Typically a telnet server running on a UNIX host uses this information to set the "TERM" environment variable. The desirable <terminal-type> values in that case are those listed in the host's /etc/termcap file or /etc/terminfo database.

Only use this command when one type of terminal is used to connect to a line; otherwise use the **session terminal** command.

Entering a line terminal <line-name> command displays the current terminal characteristics for the specified line.

```
NB:Top>Configure>Line> terminal line01
Terminal type: none
Rows: not set   Cols: not set
```

The next example shows you how to set the line's terminal type to vt100.

```
NB:Top>Configure>Line> terminal line01 vt100
NB:Top>Configure>Line> terminal line01
Terminal type: "vt100"
Rows: not set   Cols: not set
```

Note: *This command does not take effect until the next login on that line.*

To remove a terminal type parameter, use a **line terminal ""** command.

```
NB:Top>Configure>Line> terminal line01 ""
NB:Top>Configure>Line> terminal line01
Terminal type: none
Rows: not set   Cols: not set
```

The -rows and -cols parameter defines the row and column characteristics of your terminal.

```
NB:Top>Configure>Line> terminal line01 -rows 24 -cols 80
NB:Top>Configure>Line> terminal line01
Terminal type: none
Rows: 24   Cols: 80
```

line test-request <line-name>

The **line test-request** command, assigned to a line's command field by the **request-call** command, is used to call back the destination specified by the caller. You can also use this command to test the callback function for a particular destination.

The **line trace** command allows you to selectively monitor inbound and/or outbound traffic across one or more serial lines. This function is useful in tracking down modem configuration problems or determining why an attempt to connect with a remote site failed.

Entering an **line trace** command without any parameters displays the current status of tracing for all lines being traced. Entering **line trace** <line-name> command without <options ...> displays the current trace status of the specified line.

The <line-name> parameter specifies which line to trace. The <options ...> parameters defines the form and content of the trace output (refer to Table LIN-1).

Table LIN-1. Line Trace Options

Option	Description
on	Enable input, output, and status tracing for specified line.
dis[able], off	Stop tracing specified line. Note: entering an **interface** or **line trace off** command without specifying an interface or line stops all tracing.
ena[ble], resume	Resume tracing specified line.
clear, none	Clear all tracing option bits of specified line.
in[put], noin[put] out[put], noout[put]	Enable/disable tracing of inbound traffic. Enable/disable tracing of outbound traffic.
status nostatus	Show modem status. Do not show modem status.
ascii hex	Show line traffic in ASCII format. Show line traffic in 2-digit HEX format.

All trace output is sent to only one serial port or telnet session – the last one used to issue a **trace** command.

The default output is in ASCII with carriage returns shown as \r, new lines shown as \n, and control characters shown as ^X (e.g., hex 03 is displayed as ^C). Characters with the parity bit set are displayed with a \@ followed by the character.

An example of how line tracing works when starting a telnet session with a remote system is shown on the following page.

```
NB:Top>Configure>Line> trace line10 on
NB:Top>Configure>Line> trace line10
line10 enabled input output status ascii
NB:Top>Configure>Line> 08:24:09.65 line10 info: Queued line10 by ruby Dialing:
state Idle, event CHATDIAL\r
Processed line10 by ruby Dialing: state Idle, event CHATDIAL, new state
ChatDial\r
08:24:09.66 line10 sent: atdt 9-555-1234\r
08:24:13.57 line10  got: 53\r
08:24:30.29 line10  got: 13\r
08:24:30.29 line10 info: CARRIER ON\r
08:24:30.61 line10 info: Queued line10 by timer: state ChatDial, event DCD ON\r
08:24:30.62 line10 info: Processed line10 by ruby Dialing: state ChatDial,
event
 DCD ON, new state Conn Orig\r
Queued line10 by ruby Dialing: state ChatDial, event CHAT\r
Processed line10 by ruby Dialing: state Conn Orig, event CHAT, new state CHAT\r
08:24:30.79 line10  got: \r
08:24:30.79 line10  got: \n\r
08:24:30.79 line10  got: \n
08:24:30.84 line10  got: Telebit's NetBlazer Vers
08:24:30.85 line10  got: ion 1.92x2-beta\r
08:24:30.85 line10  got: \n\r
08:24:30.85 line10  got: \nruby login:
08:24:30.87 line10 sent: jade\r
08:24:30.96 line10  got: jade
08:24:30.97 line10  got: \r
08:24:30.97 line10  got: \n
08:24:31.01 line10  got: Challenge:0D8E6F2B21F36E2B:\r
08:24:31.01 line10  got: \nResponse:
08:24:31.02 line10 sent: 9300756E3F975F69\r
08:24:31.13 line10  got: 9300756E3F975F69\r
08:24:31.15 line10  got: \n\r
08:24:31.15 line10  got: \n
08:24:31.17 line10  got: Packet mode enabled\r
08:24:31.18 line10  got: \n
08:24:31.19 line10 info: Queued line10 by line10 ruby Chatting: state CHAT,
event Login\r
08:24:31.20 line10 info: Processed line10 by ruby Dialing: state CHAT, event
Login, new state Logged-In\r
```

line v25bis <line-name> on|off

The **line v25bis** <line-name> **on** command configures the specified line to use the V.25*bis* protocol. Before issuing this command, make sure that a ISDN terminal adapter is attached to the line.

Refer to the *NetBlazer Configuration Guide* for instructions on how to use a synchronous line with an ISDN terminal adapter.

Issuing a **line v25bis** <line-name> **off** command stops running the V.25*bis* protocol on the specified line.

List Command

The **list** command takes you to the List command level, which is made of various commands to display statistical information.

```
NB:Top> list
NB:Top>List> ?
Available commands:
alias           arp             asy             destination
dialout         domain          dynamic         ethernet
filter          group           history         icmp
interface       ip              line            mem
pgroup          ppp             ps              rip
route           servers         session         slip
snmp            socket          tcp             token
top>            udp             up>             user
?
```

Commands at this level are only available to users who have system status or configure privileges.

For information on using the commands listed in Table LIS-1, refer to the equivalent command description. For example, the **list route** command is described under the **ip route list** command description.

Table LIS-1. Equivalent List Commands

List Command	Equivalent to:	List Command	Equivalent to:
alias	alias list	ip	ip list
arp	arp list	line	line list
asy	asy list	ppp	ppp list
destination	destination list	rip	ip rip list
dialout	dialout list	route	ip route list
domain	ip domain list	servers	servers list
dynamic	interface dynamic list	session	session list
filter	ip filter list	slip	slip list
group	group list	snmp	snmp list
history	history list	tcp	ip tcp list
icmp	ip icmp list	user	user list
interface	interface list		

The remainder of this section documents those **list** commands that are not documented elsewhere in this manual.

list ethernet

The **list ethernet** command displays the status of the Ethernet adapter boards installed in the NetBlazer.

list mem

The **list mem** command displays the NetBlazer memory usage status. This information is primarily used for debugging purposes.

list pgroup

The **list pgroup** command displays the process groups: e.g., processes that are grouped together by user (a user may have more than one process running at a time). This information is primarily used for debugging purposes.

list ps

The **list ps** command shows all the processes running in the NetBlazer. This information is primarily used for debugging purposes.

list slip

The **list slip** command displays SLIP statistics.

list socket [<number>]

The **list socket** command shows all the sockets (logical connections between transport services and applications) in the NetBlazer. This information is primarily used for debugging purposes.

list udp

The **list udp** command displays UDP statistics.

Logout Command

The **logout** command logs you out of the NetBlazer. Before logging out, enter a **session list** command to verify that you do not have any open sessions.

```
NB:Top> session list
 #    S#  Type     Rcv-Q Snd-Q State        Remote socket     Pgroup
NB:Top> logout


Telebit's NetBlazer Version 1.92x2-beta

NB login:
```

If you issue this command from a terminal connected directly to the NetBlazer, the system displays the login prompt to allow another user to log into the NetBlazer.

If you are logged in to the NetBlazer through a telnet, the telnet session is closed. If you are logged in to the NetBlazer through a modem connection, the NetBlazer instructs the modem to disconnect and hang up the telephone line.

Message Command

The **message** command takes you to the Configure Message command level. To view a list of the commands available at this level, type a question mark (?) at the Configure Message command level.

```
NB:Top> message
NB:Top>Configure>Message> ?
Available commands:
hello           list            login           shutdown
top>            up>             ?
```

Commands at this level are used to enter messages that are sent to users who are logged in to the NetBlazer.

message hello [<message>]

The **message hello** command allows you to enter a greeting message that is displayed before the login prompt. You can enter a single-line message or a multiline message consisting of up to 1024 characters.

The following example shows you how to enter a single-line message.

```
NB:Top>Configure>Message> hello 'Telebit's NetBlazer Version %s'
```

The next example shows you how to enter a multiline message.

```
NB:Top>Configure>Message> hello
Enter the message for 'hello' (up to 990 characters).
Type . at the start of a line to end the message.
Line 1: Telebit Corporation Publications Department
Line 2: NetBlazer Version %s
Line 3: Connected to %s from %s
Line 4: .
```

The %s in both of the above examples is a special variable that is replaced by the current system version number, host name, and port (in that sequence) when the message is displayed.

message list

The **message list** command displays the current contents of the hello, login, and shutdown messages.

message login [<message>]

The **message login** command is used to enter a message that is presented to every user after they log into the NetBlazer. This is similar to the motd (message of the day) command used by UNIX systems.

You can enter a single-line message (enclosed in quotes) or a multiline message consisting of up to 1024 characters. An example of a multiline message follows.

```
NB:Top>Configure>Message> login
Enter the message for 'login' (up to 990 characters).
Type . at the start of a line to end the message.
Line 1: Telebit Proprietary Information
Line 2: Unauthorized Use is Prohibited.
Line 3: .
```

message shutdown [<message>]

The **message shutdown** command is used to enter a message that is sent to every user a specified number of seconds before the NetBlazer is shut down by a **shutdown** command.

You can enter a single-line message (enclosed in quotes) or a multiline message consisting of up to 1024 characters. An example of a multiline message follows.

```
NB:Top>Configure>Message> shutdown
Enter the message for 'shutdown' (up to 990 characters).
Type . at the start of a line to end the message.
Line 1: *** WARNING ***
Line 2: The NetBlazer will be shut down in %s seconds.
Line 3: Please logout
Line 4: .
```

The %s in the above example is a required variable that is substituted by the <interval in seconds> parameter in the **shutdown** command.

When entering a single-line message that contains an exclamation point, enter it as \! to avoid confusion with a history reference.

Msglog Command

The **msglog** <message> command is used to send a message to *syslogd* on the host system specified by the **syslog host** command. This command is useful for verifying that messages from the NetBlazer are reaching the appropriate log file on the *syslog* host system.

The <message> parameter may be a single word or several words enclosed in quotes such as: "Test message from the NetBlazer."

If the message sent by **msglog** does not reach the host log file, refer to the **syslog host** command description for troubleshooting instructions.

Phonelist Commands

The **phonelist** command takes you to the Configure Phonelist command level. To view a list of the commands available at this level, type a question mark (?) at the Configure Phonelist command level.

```
NB:Top> phonelist
NB:Top>Configure>Phonelist> ?
Available commands:
delete          deny            flush           permit
list            test            top>            up>
?
```

Commands available at this level maintain a list of telephone numbers that is used to accept or reject an incoming call by matching the phone number provided by an ISDN terminal adapter attached to the syn0 line.

Initially the phone list is empty, and all incoming calls are accepted. When the first telephone number is added to the list by either the phonelist deny or permit commands, the default is changed to the opposite state. For example, if the first phone number added to the list is permitted, all other phone numbers are rejected.

phonelist delete <phonenumber...>

The **phonelist delete** command removes one or more telephone numbers from the phone list.

phonelist deny default | <phonenumber ...>

The **phonelist deny** command specifies one or more telephone numbers from which the NetBlazer will not accept a call through an ISDN terminal adapter.

If the first number added to the phone list is added with a **deny** command, the default is to permit all other phone numbers not specifically denied.

If you enter **phonelist deny default**, all numbers that are not specifically permitted in the list are denied.

phonelist flush

The **phonelist flush** command removes all entries from the phone list except the default entry.

phonelist permit default | <phonenumber ...>

The **phonelist permit** command specifies one or more telephone numbers from which the NetBlazer will accept a call through an ISDN terminal adapter.

If the first number added to the phone list is added with a **permit** command, the default is to deny all other phone numbers not specifically permitted.

If you enter **phonelist permit default**, all numbers that are not specifically denied in the list are permitted.

phonelist list

The **phonelist list** command displays the phone number list.

phonelist test <phonenumber> [<phonenumber> ...]

The **phonelist test** command checks the permit/deny status of one or more phone numbers.

PPP Commands

The **ppp** command takes you to the Configure PPP command level. To view a list of the commands available at this level, type a question mark (?) at the Configure PPP command level.

```
NB:Top> ppp
NB:Top>Configure>PPP> ?
Available commands:
activeopen      close           list            options
passiveopen     protocols       pool            reset
timeout         top>            trace           up>
?
```

These commands control the operation of PPP interfaces.

ppp activeopen <interface>

The **ppp activeopen** command starts PPP running on the specified interface. This command must be issued before any packets can be sent through the interface. When created, PPP interfaces default to the active state.

ppp close <interface>

The **ppp close** command terminates the link specified by <interface>. This stops the flow of packets over the link until the **ppp activeopen** command is again issued.

Note that a remote interface in the closed state can be opened only by the **activeopen** command or by rebooting the NetBlazer.

ppp list [<interface>]

The **ppp list** command displays information about the PPP interfaces. If the <interface> parameter is excluded, information about all PPP interfaces is displayed. If the <interface> is specified, detailed information about that interface is displayed.

ppp options <interface> <protocol> [<option> on|off]

This command allows you to enable or disable the PPP protocol options used in negotiating PPP connections, assuming the PPP protocols are turned on (see **PPP protocol** command). These options may be enabled or disabled on a per interface and protocol (IPCP or IPXCP) basis.

The options for each protocol are as follows:

IPCP Option	Description	Default
compress	Van Jacobsen Compression	on
address	Address (Option 3 or IPCP Specifications)	off
oldaddress	Old Address (Option 1 of IPCP Specification)	off

IPXCP Option	Description	Default
compress	Telebit's IPX header compression (CIPX*)	on
net	Network Number	off
node	Node Number	off
*CIPX is currently an Internet draft written by Telebit		

Specifying only the protocol parameter returns the current setting of the options for that protocol. If you supply the option, you must also specify on or off.

ppp passiveopen <interface>

The **ppp passiveopen** command opens the specified interface in a passive state. This allows the other side of the link to start a PPP connection via the **ppp activeopen** command.

ppp protocols <interface> [<protocol> on|off]

The **ppp protocols** command enables or disables the IPCP and IPXCP protocol options. To determine the state of the protocols, specify only the interface, and the state of all protocols is shown. To change the state of a protocol, specify the <interface>, the <protocol>, and on or off.

The **ppp pool** command supports the capability of dynamically assigning IP addresses through PPP address negotiation. Once set up, when a user dials in to the NetBlazer (using PPP software), the NetBlazer assigns the remote user an IP address.

A peer typically calls in to the NetBlazer and attempts to negotiate its address by specifying a 0.0.0.0 host address. This negotiation uses IPCP option 3 (address), so this option must be on.

Assuming an address pool has been established for the interface, the NetBlazer assigns one. If negotiations are successfully complete, both ends know their respective addresses, and the IPCP link comes up. In the process, the NetBlazer adds an host auto-route to the routing table for the assigned address. When the interface goes away, the auto-route is removed from the routing table.

To determine if a PPP address pool as been established for interface, issue the **pool** command as follows:

```
NB:Top>Configure>PPP> pool yoyo
No PPP address pool
```

To define a single address pool, specify the interface and just the address.

```
NB:Top>Configure>PPP> pool yoyo 143.191.64.49
Enabled IPCP address negotiation for yoyo
NB:Top>Configure>PPP> pool yoyo
PPP address pool 143.191.64.49 thru 143.191.64.49
```

To define a range of addresses, specify the lowest IP address and a count.

```
NB:Top>Configure>PPP> pool yoyo 143.191.64.49 5
Enabled IPCP address negotiation for yoyo
NB:Top>Configure>PPP> pool yoyo
PPP address pool 143.191.64.49 thru 143.191.64.53
```

To clear the pool of addresses, issue **pool** <interface> **0** command.

```
NB:Top>Configure>PPP> pool yoyo 0
Cleared PPP address pool for yoyo
```

Users may assign the same pool of address to more than one interface. The NetBlazer checks to see if an address is already in use.

Note: *If you wish to establish an address pool available to a number of users, it is necessary to create multiple interfaces. The same pool may then be assigned to all the interfaces.*

Tip: *The most convenient way to establish multiple interfaces is by using the NetBlazer's* ***history*** *substitution command.*

```
NB:Top>Configure>PPP> pool yoyo 143.191.64.49 5
Enabled IPCP address negotiation for yoyo
NB:Top>Configure>PPP> ^yoyo^gyro^
pool gyro 143.191.64.49 5
Enabled IPCP address negotiation for gyro
```

ppp reset <interface>

The **ppp reset** command resets the link specified by <interface> and brings it back up. This command negotiates the link down and brings it back up to the state in which it started (active or passive).

ppp timeout <interface> [<seconds>]

The **ppp timeout** command sets the maximum timeout period for the PPP interface (form PPP negotiation request to response from the remote host or user) over the asynchronous and synchronous lines.

For asynchronous line, the default is 5 seconds before timeout. For synchronous lines, the default is 3 seconds.

ppp trace <interface> [all lcp ipcp atcp ipxcp off]

The **ppp trace** command is a diagnostic aid useful in troubleshooting PPP negotiation problems. PPP trace syslogs to a host, a buffer, a file, a session, or the console. You must first set up syslog in order to see output (see the **syslog** command description).

The **ppp trace** command options include all lcp ipcp atcp ipxcp off, where lcp is Link Control Protocol, ipcp is IP Control Protocol, atcp is AppleTalk Control Protocol, and ipxcp is IPX Control Protocol.

The **ppp trace** command also operates on dynamic PPP interfaces that are currently down. You can set ppp tracing for such an interface and when it comes up, the interface is traced as specified.

Note that the **ppp trace** command is not saved to the STARTUP.CNF file. It is active only until turned off or until the NetBlazer is rebooted.

Reboot Command

The **reboot** [fast] command restarts the NetBlazer using the configuration information saved in the STARTUP.CNF file. If this command is entered without the **fast** option, the NetBlazer attempts to close any sessions in progress before rebooting the system. A **reboot fast** command immediately reboots the system without attempting to close the sessions.

The **shutdown** command is a more friendly way to reboot the NetBlazer in that it notifies users that you are about to reboot the NetBlazer and gives them a chance to finish what they are doing before the reboot.

Before issuing a **reboot** command, verify that you have a NetBlazer system diskette installed. Also, make sure that there are no other users logged in to the NetBlazer by issuing a **who** command. If users are logged in, you can send them a message informing them that you are about to reboot the NetBlazer by issuing a **write** or **wall** command.

Save Command

The **save** [local|remote] [<startup>] [<passwd>] command saves the current configuration and user records on the NetBlazer system diskette or on a remote TCP/IP host.

Any time you modify the NetBlazer configuration, make sure that you have saved the current settings and user records in the appropriate location.

save

Issuing a **save** command with no parameters saves the current configuration settings and user records to the STARTUP.CNF and PASSWD file on the NetBlazer's system diskette and warns you if these files should also be saved remotely.

save local

The **save local** command saves the current configuration and user records to the STARTUP.CNF and PASSWD file on the NetBlazer's system diskette, regardless of the original source of the file.

save remote

The **save remote** command saves the current configuration and user records in the remote files specified by the **files startup** and **files passwd** command settings. Because this command uses TFTP to write the files to a remote host, the remote files must have read/write permissions set for everyone.

If allowing everyone to write to a startup or passwd file presents a security problem for your site, consider saving the file locally on the NetBlazer disk and then using FTP to transfer the file to the remote host.

save <startup> <passwd>

The **save** <startup> <passwd> command saves the current configuration settings and user records in the files specified by the <startup> <passwd> parameters. To save a file on a remote system, enter the file name as <host>:<pathname> where <host> is the domain name or IP address of the remote host, and <pathname> is the full path name of the file. Remote files must have read/write permissions set for everyone.

If you only want to save the user records, enter a **save /dev/null** <passwd> command. Likewise, if you only want to save the current configuration settings, enter a **save** <startup> **/dev/null** command.

Security Commands

The **security** commands restrict the user's write, dial, and talk privileges and controls whether or not a password prompt is issued when an a unauthorized user attempts to log in.

To view a list of the commands available at this level, type a question mark (?) at the Configure Security command level.

```
NB:Top>Sessions> security
NB:Top>Configure>Security> ?
Available commands:
login           numberdial      top>            up>
usertalk        write           ?
```

security login [enable|disable]

The **security login** command enables or disables the prompt for a password when a user attempts to log in with an unauthorized user ID.

Entering this command without specifying enable or disable shows the current setting.

```
NB:Top>Configure>Security> login
Security login is disabled
```

The NetBlazer is shipped with **security login** disabled, which means that a password prompt is not displayed when a user attempts to log in with an invalid user ID.

If you enable the security login function as shown below, the password prompt is displayed even when the user has entered an invalid user ID.

```
NB:Top>Configure>Security> login enable
NB:Top>Configure>Security> login
Security login is enabled
```

The user is given another opportunity to enter a valid user ID and password.

Entering a security login disable command disables the prompt for a password when a user attempts to log in with an unauthorized user ID.

```
NB:Top>Configure>Security> login disable
NB:Top>Configure>Security> login
Security login is disabled
```

security numberdial [enable|disable]

The **security numberdial** command controls whether or not logged-in users are allowed to issue a **session dial** <phonenumber> command.

Entering this command without specifying enable or disable shows the current setting.

```
NB:Top>Configure>Security> numberdial
No one may dial phone numbers directly
```

The NetBlazer is shipped with **security numberdial** disabled, which means that users cannot dial a telephone number directly.

If you issue a **security numberdial enable** command, all users can use the **session dial** <phonenumber> command to dial a telephone number.

```
NB:Top>Configure>Security> numberdial enable
NB:Top>Configure>Security> numberdial
Anyone may dial phone numbers directly
```

Entering a **security numberdial disable** command prevents all users from dialing a telephone number directly.

```
NB:Top>Configure>Security> numberdial disable
NB:Top>Configure>Security> numberdial
No one may dial phone numbers directly
```

security usertalk [enable|disable]

The **security usertalk** command enables or disables the **session talk** command for users who do not have configuration privileges.

Entering this command without specifying enable or disable shows the current setting.

```
NB:Top>Configure>Security> usertalk
Only users with configuration privileges may talk to a line
```

The NetBlazer is shipped with **security usertalk** disabled – only users with configuration privileges can issue a **session talk** command.

If you issue a **security usertalk enable** command, the **session talk** command can be used by all users.

```
NB:Top>Configure>Security> usertalk enable
NB:Top>Configure>Security> usertalk
Anyone may talk to a line
```

Entering a **security usertalk disable** command prevents users without configuration privileges from issuing a **session talk** command.

```
NB:Top>Configure>Security> usertalk disable
NB:Top>Configure>Security> usertalk
Only users with configuration privileges may talk to a line
```

security write [enable|disable] [user|priv|status]

The **security write** command controls whether or not a group of users is allowed to use the **write** command based on their privilege level.

Entering a **security write** command without any parameters shows the current write permissions for each group.

```
NB:Top>Configure>Security> write
User       :enabled
StatPriv   :enabled
ConfPriv   :enabled
```

The NetBlazer is shipped with the **write** command enabled for all groups of users. If you do not want normal users to use the **write** command enter a **write disable user** command at the Configure Security level.

```
NB:Top>Configure>Security> write disable user
NB:Top>Configure>Security> write
User       :disabled
StatPriv   :enabled
ConfPriv   :enabled
```

The next example shows you how to enable write permissions for normal users.

```
NB:Top>Configure>Security> write enable user
NB:Top>Configure>Security> write
User       :enabled
StatPriv   :enabled
ConfPriv   :enabled
```

Servers Commands

The **servers** command takes you to the Configure Servers command level where you can start or stop various servers in the NetBlazer. To view a list of the commands available at this level, type a question mark (?) at the Configure Servers command level.

```
NB:Top> server
NB:Top>Configure>Servers> ?
Available commands:
list              start>          stop>           top>
up>               ?
```

servers list

The **servers list** command displays the current status of the NetBlazer servers. The various servers and their parameters are listed alphabetically with the **servers start** command.

servers start <server-name> [<arguments>]

The **servers start** command starts a server specified by the <server-name> parameter. See the server commands listed alphabetically below for descriptions of the servers and their arguments.

servers start adsp

The **adsp** command starts or stops the AppleTalk Data Stream Protocol (ADSP) server. This server allows an interactive session with a remote host. ADSP is a sessions-level protocol in the AppleTalk protocol stack. ADSP is similar to IP's TELNET.

The default is off. To start the server, enter a **server start adsp** command. Stop the server with the **server stop adsp** command.

Note: *An AppleTalk interface must exist before the* adsp *server appears on the* servers list.

servers start bootpd <destination>

The **bootpd** command starts or stops the NetBlazer BOOTP server. This server forwards BOOTP requests from another NetBlazer to a specified IP host.

The default is off. To start the server, enter a **server start bootpd** <destination> command, where <destination> is the IP address or host name of the host BOOTP server. Stop the server with the **server stop bootpd** command.

Refer to the Network Management chapter in the *NetBlazer Configuration Guide* for instructions on how boot a NetBlazer over the network.

servers start discard [<port-number>]

The **discard** command starts or stops the discard server. This server is used for network testing. When this server is turned on, the NetBlazer accepts and acknowledges data received on TCP port 9. The data is discarded.

The default is off. To start the server, enter a **server start discard** <port-number> command. The NetBlazer's default discard port number is 9. You can use a different port number when you enter a **server start discard** <port-number> command. Stop the server with the **server stop discard** command.

servers start echo [<port-number>]

The **echo** command starts or stops the echo server. This server is used for network testing. When this server is turned on, the NetBlazer accepts and acknowledges TCP segments received at port 7. The data received by the NetBlazer is echoed back to the original source.

The default is off. To start the server, enter a **server start echo** <port-number> command. The NetBlazer's default echo port number is 7. You can use a different port number when you enter a **server start echo** <port-number> command. Stop the server with the **server stop echo** command.

servers start finger [<port-number>]

The **finger** command stops or starts the finger server. This server enables other hosts to query the NetBlazer for information about NetBlazer users using the finger protocol.

The default is on. To stop the server, enter a **server stop finger** command. Start the server with the **server start finger** <port-number> command. The NetBlazer's default FTP port number is 79. You can use a different port number when you enter a **server start finger** <port-number> command.

servers start ftp [<port-number>]

The **ftp** command stops or starts the File Transfer Protocol (FTP) server. When the FTP server is running, the system administrator may use FTP to automatically transfer new software and configuration files to the NetBlazer from a remote host.

The default is on. To stop the server, enter a **server stop ftp** command. Start the server with the **server start ftp** <port-number> command. The NetBlazer's default FTP port number is 21. You can use a different port number when you enter a **server start ftp** <port-number> command.

servers start gdb [<port-number>]

The **gdb** command starts or stops the GDB server. This server lets Telebit engineers do remote debugging over a network. The GDB server is off unless Telebit engineers are debugging your NetBlazer.

servers start ipxsnmp [<port-number>]

The **ipxsnmp** command starts or stops the IPX SNMP server. This server is used for IPX network management tasks.

The default is off. To start the server, enter a **server start ipxsnmp** <port-number> command. The NetBlazer's default IPX SNMP port number is 900Fh. You can use a different port number when you enter a **server start ipxsnmp** <port-number> command. Stop the server with the **server stop ipxsnmp** command.

servers start raw

The **raw** command starts or stops the raw server. This server allows you to establish a raw TCP connection from a host to the modem or other asynchronous device attached to a NetBlazer line.

The default is off. To start the server, enter a **server start raw** command. Stop the server with the **server stop raw** command.

servers start rip

The **rip** command stops or starts the IP RIP server. This server is used by the NetBlazer to learn routes to other IP networks from IP RIP advertisements sent by other routers on the IP network.

The default is on. To stop the server, enter a **server stop rip** command. Start the server with the **server start rip** command.

servers start snmp [<port-number>]

The **snmp** command stops or starts the Simple Network Management Protocol (SNMP) server. This server is used for IP network management tasks.

The default is on. To stop the server, enter a **server stop snmp** command. Start the server with the **server start snmp** <port-number> command. The NetBlazer's default SNMP port number is 161. You can use a different port number when you enter a **server start snmp** <port-number> command.

servers start status

The **status** command starts or stops the status server. This server sends status messages to an internal console. To use the status server, you must install an internal monitor card in the NetBlazer.

The default is off. To start the server, enter a **server start status** command. Stop the server with the **server stop status** command.

servers start telnet [<port-number>]

The **telnet** command stops or starts the TELNET server. This server allows users and administrators to log into the NetBlazer remotely through the network.

The default is on. To stop the server, enter a **server stop telnet** command. Start the server with the **server start telnet** <port-number> command. The NetBlazer's default TELNET port number is 23. You can use a different port number when you enter a **server start telnet** <port-number> command.

servers start watchdog <interface> [<seconds>]

The **watchdog** command starts or stops the watchdog server. This server sends PPP echo requests to detect if a dedicated PPP interface goes down. This allows you to configure the NetBlazer to reroute packets through a dynamic interface when a dedicated interface goes down.

The default is off. To start the server, enter a **server start watchdog** <interface> [<seconds>] command, where <interface> is the dedicated interface you want watched and <seconds> is the interval between PPP echo requests. The <seconds> interval must be equal to or greater than 5 seconds. If you do not enter a <seconds> parameter, the NetBlazer defaults to 120 seconds between PPP echo requests.

Note: *For certain applications, the 120 second default value may be too long. Applications that require a faster switchover to the backup interface may require a shorter value. Few applications will require values shorter than 20 to 30 seconds. Note that shorter intervals increase overhead traffic on the interface.*

Stop the server with the **server stop watchdog** command.

Refer to the Dedicated Interface Configuration chapter in the *NetBlazer Configuration Guide* for information on how to set up a dynamic backup interface for a dedicated PPP interface.

servers stop <server-name>

The **servers stop** command is used to disable a server that has been started by a previous start command. The <server-name> parameter specifies which server to stop. Servers may be started and stopped at any time, but be aware that if you are connected to the NetBlazer through a telnet session, there may be no way to restart the telnet server without resetting the NetBlazer.

Sessions Commands

The **sessions** command takes you to the Sessions command level. To view a list of the commands available at this level, type a question mark (?) at the Sessions command level.

```
NB:Top> session
NB:Top>Sessions> ?
Available commands:
abort           bang            binary          chatdial
close           connect         continue        dial
disconnect      eol             escape          fg
finger          ftp             jobs            kill
list            ping            rlogin          send
talk            telnet          terminal        tftp
top>            up>             ?
```

Commands at this level allow authorized users to start, stop, and control communication sessions with local or remote machines.

session abort [<session-number>]

The **session abort** command aborts an FTP data transfer operation. It is the only command permitted during an FTP session. If no <session-number> parameter is entered, the NetBlazer displays a list of active sessions and prompts you for the session number.

session bang <host> [<length> <count> <interval> <outstanding>]

The **session bang** command sends ICMP echo requests to the specified remote host. This command is equivalent to a **session ping** command except that instead of printing a line of statistics for each response, **session bang** only prints a single character (!). Lost or out of sequence packets are shown as a dot (.), and ICMP messages are shown as an X in the **bang** display. Refer to the **session ping** command for an explanation of the parameters used by both these commands.

session binary <session-number> [on|off]

The **session binary** command turns on or off the telnet binary option. This allows the NetBlazer to transmit and receive all 256 possible character codes. This is useful for such operations as X-terminals and DEC VT-200 series terminals. If binary mode is not enabled, the NetBlazer will transfer only a subset of the 7-bit ASCII code.

If the on or off parameter is omitted, the current telnet binary setting for the specified session number is displayed.

session chatdial <site-name>

The **session chatdial** <site-name> command establishes a character mode connection with a specified remote site and uses the chat script assigned to the site's dialout record to log in. The NetBlazer looks up the <site-name> in the dialout list to get the information needed to make the connection. Refer to the **dialout** command descriptions for instructions on setting up dial-out destinations.

session close [<session-number>]

The **session close** command closes the session specified by the <session-number> parameter. This is the preferred way to terminate a session. If no <session-number> is entered, the NetBlazer displays a list of active sessions and prompts you for the session number.

session connect <destination-name>|<destination-number>

The **session connect** command starts a telnet or dial-out session with a pre-defined destination. The NetBlazer first looks up the destination name in the user's assigned destination group to get the connection mode (telnet or dial) and address (domain name, IP address, or telephone number) and then makes the appropriate connection.

You can enter a destination number assigned to a destination in place of the destination name parameter. To get a list of destination numbers and names, press the return key when you are at the NetBlazer Sessions command level.

```
Top>Sessions> press the Return key
Accessible sites:
1  marketing          Marketing Computer
2  databank           Data Bank Service
```

The above example shows that two sites, marketing and databank, are accessible through the **connect** command for this user. Entering **connect 1** is equivalent to entering **connect marketing**. If you get an "All sites are accessible" message, you probably do not have a destination group assigned to your user ID account.

For more information about setting up destinations used by the **connect** command, see the **destination** command descriptions.

session continue [<session-number>]

The **session continue** command returns to the session specified by the <session-number> parameter. If no <session-number> parameter is entered, the NetBlazer prompts you for the session number.

You also can use the **session fg** command to continue a session.

session dial <site-name>|<phone-number>

The **session dial** <site-name> command establishes a character mode connection with a specified remote site. The NetBlazer looks up the <site-name> in the dialout list to get the information needed to make the connection. Refer to the **dialout** command descriptions for instructions on setting up dial-out destinations.

To allow dialing a telephone number directly, the system administrator must enter a **security numberdial enable** command.

session disconnect [<session-number>]

The **session disconnect** command terminates the session specified by the <session-number> parameter. It is intended for dial-out connections, but it can be used to close any session. If no <session-number> parameter is entered, the NetBlazer prompts you for the session number.

session eol [standard | unix]

The **session eol** command selects the proper end-of-line processing during a telnet session when the operator presses the return key. The default setting of "standard" should be used for most systems. The "unix" option should only be used when the remote system fails to recognize the carriage return key as indicating end-of-line.

session escape [<character>]

The **session escape** command displays or changes the keystroke used in a escape sequence to interrupt a session and get back to the command prompt. The escape sequence consists of the escape character followed by a carriage return.

Changing the escape character on the NetBlazer is desirable, for example, when telnet on your computer uses the same escape character as the NetBlazer. In this case, telnet traps the escape character that may have been intended for the NetBlazer. By changing the escape character on the NetBlazer to a value that does not match the telnet escape character, telnet allows the escape character to pass through to the NetBlazer.

Entering the command without the <character> parameter displays the current escape character, normally ctrl-].

The optional <character> parameter specifies which control character should be used as the escape character. Be sure to enter the desired keystroke exactly as you would use it. For example, if you decide to change the escape character to ctrl-/, you should hold down the control key and press the / key.

The escape character is set on a per session basis. The escape character used by one user may differ from the escape character used by other users logged into the system.

If a user with configuration privileges issues a **configure save** command after changing the escape character, that character becomes the default escape character when the NetBlazer is rebooted. The escape character setting for all other users only applies to their current login session.

session fg [<session-number>]

Like the **session continue** command, the **session fg** (foreground) command continues a session specified by the <session-number> parameter. If no <session-number> is entered, the NetBlazer prompts you for the session number.

session finger [<user>] [@<host>]

The **session finger** command invokes the finger client. The finger client permits a user to request information about a user at a host. The field <user> is a valid user ID on the system being queried. The field @<host> is the @ character immediately followed by an IP address or a domain name.

If <user> is specified without a host specification, the local (NetBlazer) finger server is queried about that user of the local NetBlazer. If just @<host> field is specified, the host being queried usually displays a list of possible users or a list of currently active users. This depends on the remote host and is not under the control of the NetBlazer.

Here are a couple of examples:

```
session finger julie@marketing.telebit.com
session finger @marketing.telebit.com
```

The first example returns information about the user julie on the host named marketing.telebit.com. The second example returns information about the current users at the host named marketing.telebit.com.

session ftp [<address>]

The **session ftp** command starts an FTP session with a host specified by the <address> parameter. The address is the host's IP address or domain name.

Once connected to the remote host, the standard set of FTP commands apply. These vary slightly from host to host so see the documentation for the particular remote host to which the connection is being established.

Caution: *Interrupting an FTP session while copying a file to the NetBlazer diskette might make the diskette unusable. Before starting the FTP session, make sure that you have a backup of your current system diskette. Do not interrupt the FTP session while it is copying a file to the NetBlazer diskette.*

session jobs

Like the **session list** command, the **session jobs** command displays a list of sessions currently running on the NetBlazer.

session kill [<session-number>]

The **session kill** command terminates the session specified by the <session-number> parameter without notifying the remote end. If no <session-number> is entered, the NetBlazer displays a list of active sessions and prompts you for the session number.

This command is used as a last attempt to close a session with a host that is no longer responding. Do not use this command to terminate a session with a remote host that is responding because the NetBlazer will terminate the session without telling the remote end that the connection is closed.

You should always first try to **close** a session and **kill** the session only if that fails.

session list

The **session list** command displays a list of sessions currently running on the NetBlazer. An example of a session status list is shown below:

```
NB:Top>Sessions> list
 #    S#  Type      Rcv-Q Snd-Q State        Remote socket      Pgroup
 0 $ 5   Telnet        0     0 Established  pearl (143.191.63.1:23)line17
```

The # field is the session number. An asterisk (*) next to the session number denotes the current session.

The S# field identifies the socket number. This field is use primarily for debugging purposes.

The Type field identifies the type of session; i.e., Telnet, FTP, Ping, etc.

The Rcv-Q displays the number of characters received on that session but not displayed. Continuing the session displays the stored data.

The Snd-Q displays the number of bytes transmitted but not acknowledged by the remote host.

The State field displays the current state of the TCP session. The most significant state is "Established" indicating that a connection exists with the remote host and that information may be transferred to/from that host. A state containing the word "SYN" indicates that the NetBlazer is attempting to establish a connection. A state containing the phrase ""FIN wait" or "Time wait" indicates that the session is being closed.

The Remote socket and Pgroup describe the session connection.

session ping <host> [<length> <count> <interval> <outstanding>]

The **session ping** command sends ICMP echo requests to the specified remote host. The NetBlazer then measures the amount of time required for the remote host to send a response.

The <host> parameter may be either a domain name or an IP address.

The optional <length> parameter is used to pad the request to a larger size specified in bytes. This parameter must be greater than 32 bytes (the minimum length allowed) to take effect. However, a value less than 32 can be used as placeholder in the command.

The optional <count> parameter specifies the number of ICMP requests to send to the remote host. If this parameter is not specified or 0 is entered as a placeholder, the NetBlazer continues issuing requests to the host until the ping session is stopped by pressing ctrl-C or by issuing a close, disconnect, or kill command.

The optional <interval> parameter specifies in seconds the time to wait between sending ICMP echo requests. This value may also be set to 0, and the NetBlazer will send pings with 10 ms separation (approximately 100 pings per second).

The optional <outstanding> parameter specifies the number of echo requests (pings) that may be transmitted before the NetBlazer stops and waits for their responses.

The ability to set the ping interval to 0 (10 ms between pings) and set the number of outstanding pings is a useful tool to place a significant load on a network without overloading the network.

session rlogin [-b|-8] [-noesc] [-l username] [<address>] [<port>]

The **session rlogin** command is used to establish an interactive session on remote host using current NetBlazer user ID as remote host login ID. If the remote host has an entry for NetBlazer in /etc/hosts.equiv, and the NetBlazer user ID matches the remote host user ID, the user is simply logged into remote host without requiring additional user name and password authentication on remote host.

The -b option starts a binary telnet session. This has the same result as escaping out of a rlogin session, turning on binary for that session, and then continuing the session.

The -8 option starts an rlogin session using 8-bit data in non-binary mode.

The -noesc option disables escape processing. This prevents the NetBlazer from enter command mode when an escape sequence is entered. Instead, the escape sequence is passed through to the remote end.

The -l option is used to overwrite the default user name. This is normally used to specify a different user name when logging onto a remote host that does not recognize the user's NetBlazer login.

Note: *The -noesc option may not be used if there are other sessions open for the same login (pgroup). Other sessions for the same login must be closed before an rlogin -noesc command will be accepted.*

The only way to return to the NetBlazer's command mode from an rlogin session that was started with the -noesc option is to close the session at the remote end (e.g., logout).

The optional <port> parameter is used to start a telnet with the host system using a port number other than telnet's well known port number 23.

session send <session-number> ao|ayt|brk|ec|el|esc|ga|ip|nop|synch

The **session send** command sends the special character sequences defined below to a remote host.

ao	Sends the telnet ao (abort output) sequence, which should cause the remote system to flush all output from the remote system to the user's terminal.
ayt	Sends the telnet ayt (are you there) sequence, to which the remote system may or may not choose to respond.
brk	Sends the telnet brk (break) sequence, which may have significance to the remote system.
ec	Sends the telnet ec (erase character) sequence, which should cause the remote system to erase the last character entered.
el	Sends the telnet el (erase line) sequence, which should cause the remote system to erase the line currently being entered.
esc	Sends the current telnet escape character. The CR or LF that would be expected by a NetBlazer as part of the escape sequence are not sent by this command; they may be sent by the user if required.
ga	Sends the telnet ga (go ahead) sequence, which likely has no significance to the remote system.
ip	Sends the telnet ip (interrupt process) sequence, which should cause the remote system to abort the currently running process.
nop	Sends the telnet nop (no operation) sequence.
synch	Sends the telnet synch sequence. This sequence causes the remote system to discard all previously typed but not yet read input. This sequence is sent as TCP urgent data and may not work if the remote system is a 4.2 BSD system – if it does not work, a lower case "r" may be echoed on the terminal.

session talk [<line-name>]

The **session talk** command is for debugging purposes. It allows you to connect to an asynchronous line and issue modem commands to an attached modem. The <line-name> parameter must be one of the lines listed by a **line list** command that is connected to a modem and not in use.

By default, only users with system configuration privileges can use the talk command. However, entering a **security user talk enable** allows all users to talk to a line.

To stop talking to the modem, send the escape character (control-]) to the NetBlazer and issue a **close**, **disconnect**, or **kill** command.

session telnet [-b|-8|-raw] [-noesc] [<address>] [<port>]

The **session telnet** command is used to establish a telnet session with a host specified by the <address> parameter. The address is the host's IP address or domain name.

The -b option starts a binary telnet session. This has the same result as escaping out of a telnet session, turning on binary for that session, and then continuing the session.

The -8 option starts a telnet session using 8-bit data in nonbinary mode.

The -raw option implies the -b and -noesc options and eliminates all telnet IAC, CR/LF/NUL, option, and suboption processing.

The -noesc option disables escape processing. This prevents the NetBlazer from enter command mode when an escape sequence is entered. Instead, the escape sequence is passed through to the remote end.

Note: *The -noesc option may not be used if other sessions are open for the same login (pgroup). Other sessions for the same login must be closed before a telnet -noesc command is accepted.*

The only way to return to the NetBlazer's command mode from a telnet session that was started with the -noesc option is to close the session at the remote end (e.g., logout).

The optional <port> parameter is used to start a telnet with the host system using a port number other than telnet's well-known port number 23.

session terminal [<terminal-type>] [-rows <rows>] [-cols <cols>]

The **session terminal** command sets or displays the terminal characteristics for the current login. Typically a telnet server running on a UNIX host uses this information to set the "TERM" environment variable. The desirable <terminal-type> values in that case are those listed in the host's /etc/termcap file or /etc/terminfo database.

Entering a session terminal command without any parameters displays the current login terminal characteristics.

```
NB:Top>Sessions> terminal
Terminal type: none
Rows: not set   Cols: not set
```

The next example shows you how to set the login terminal type to vt100.

```
NB:Top>Sessions>  terminal vt100
NB:Top>Sessions> terminal
Terminal type: "vt100"
Rows: not set   Cols: not set
```

To remove a terminal type parameter, use a **session terminal ""** command.

```
NB:Top>Sessions> terminal ""
NB:Top>Sessions> terminal
Terminal type: none
Rows: not set   Cols: not set
```

The -rows and -cols parameter defines the row and column characteristics of your terminal.

```
NB:Top>Sessions> terminal -rows 24 -cols 80
NB:Top>Sessions> terminal
Terminal type: none
Rows: 24   Cols: 80
```

session tftp [<address>]

The **session tftp** command starts an TFTP session with a host specified by the <address> parameter. The address is the host's IP address or domain name.

Once connected to the remote host, the standard set of TFTP commands apply. These vary slightly from host to host so see the documentation for the particular remote host to which the connection is being established.

Caution: *Interrupting an TFTP session while copying a file to the NetBlazer diskette might make the diskette unusable. Before starting the TFTP session, make sure that you have a backup of your current system diskette. Do not interrupt the TFTP session while it is copying a file to the NetBlazer diskette.*

Setconsole Command

The **setconsole** [internal|<line-name>] command selects the line where the NetBlazer displays debugging information.

If **setconsole internal** is entered, the debugging messages are sent to an internal monochrome display adapter if one is present.

If **setconsole** <line-name> is entered, the debugging messages are displayed on a terminal attached to that line (line00, line01, etc.).

Note: *The NetBlazer is shipped from the factory configured to send console messages to line00 or line02, which are intended to be used only with an asynchronous terminal. Modems should be connected to the multiport lines, then to line01. If you must attach a modem to line00 or line02, issue a* ***setconsole internal*** *command to the NetBlazer in order to prevent console messages from adversely affecting the operation of the modem.*

Shutdown Command

The **shutdown** [<interval in seconds>] command sends a shutdown message to all users warning them that the NetBlazer will be shut down in the number of seconds specified by the [<interval in seconds>] parameter.

```
NB:Top> shutdown 60

Message from root on 192.124.95.18:11329

****System will go down in 60 seconds!!****

Message from root on 192.124.95.18:11329

****System will go down in 30 seconds!!****

Message from root on 192.124.95.18:11329

****System will go down in 15 seconds!!****

Message from root on 192.124.95.18:11329

****System will go down in 7 seconds!!****

Rebooting will immediately take down the system.
passwd has been changed. Do you want to save it? (y/n)y
There are no other users logged on.
Are you sure you want to do this? (y/n)
```

Notice in the above example that the NetBlazer repeats the shutdown message in half-live intervals. When the shutdown period has timed out, the NetBlazer gives you an opportunity to save the password file if it has been changed since the last save and then reboot or abort the shutdown process.

If you do not specify a shutdown interval, the NetBlazer uses the default value of 20 seconds. The message shown in the above example is the NetBlazer's default message. You can change the message sent by the shutdown command via the **message shutdown** command.

SNMP Commands

The **snmp** command takes you to the Configure SNMP command level. To view a list of the commands available at this level, type a question mark (?) at the Configure SNMP command level.

```
NB:Top> snmp
Available commands:
authentication  community>      enterprise      list
startup         sysContact      sysLocation     sysName
top>            up>             ?
```

This set of commands controls the configuration of the SNMP (Simple Network Management Protocol) server. These commands are normally used to configure the NetBlazer to be managed by one or more SNMP management stations through which:

- SNMP commands may be issued to read or set values in the NetBlazer standard MIB and enterprise MIB;
- Authentication traps can be switched on or off. These traps are used to report receipt of an invalid community group name attempting to issue an SNMP command;
- Administrative information can be obtained, including system name, contact (name and phone number), and its physical location.

Community Groups and Security

Before a management station can issue an SNMP command to a NetBlazer, the NetBlazer must be configured to recognize one or more management stations. This is accomplished by establishing an SNMP community name and a list which contains the host names of all management stations through which SNMP access to the NetBlazer is desired.

It is also possible to admit all hosts by not entering any host names. Community names are thus somewhat like passwords. SNMP commands can directly access (get or set) objects and variables in the NetBlazer's MIB-II and enterprise MIB.

Trap Community

A special community may be designated to receive authentication and startup trap information. This community is established using the normal community add operation by specifying **neither** read or write as the access option. Only one trap community is permitted at any given time.

An authentication trap is sent when a query is received that has an invalid community name, or the community has a host list defined and the sending host is not on the list.

Startup traps are sent went a NetBlazer is initially started (cold boot), or reset (warm boot).

Privileges

Three levels of SNMP access privileges to the NetBlazer's enterprise MIB are provided:

1. Read only
2. Read/Write
3. The Enterprise MIB switched off

Note: *When the NetBlazer is initially powered on, the default community setting permits any host to access the Enterprise MIB. In other words, if no hosts are listed, all hosts are accepted. Be sure to set up your community lists and hosts as part of the initial configuration.*

The Telebit Enterprise MIB

Because the configurability in MIB-II is limited (with respect to the NetBlazer's functionality), an enterprise was developed to provide extended configurability. This approach provides nearly complete configurability of the NetBlazer for both Telebit's Configuration Graphic User Interface utility, and network management stations.

System Administrators wishing to directly access the Telebit enterprise MIB using their SNMP network management software are directed to the text files embedded in the Telebit enterprise MIB. This file is on the Utility distribution diskette, and is heavily commented to provide adequate information for persons knowledgable in integrating managed object MIBs into network management applications.

The **snmp enterprise** command is provided to either enable or disable SNMP access to the enterprise MIB.

Note:

1. *If you plan to use the Telebit Configuration GUI Utility, it is necessary that the Enterprise MIB be* enabled *with read|write permission.*
2. *If you plan to use the Telebit enterprise MIB with a network management station, refer to the documentation provided with your MIB compiler or contact your vendor for more information about how to use the Telebit MIB with a third party MIB compiler. You may also contact Telebit Tech Support at 1-800-TELEBIT.*

SNMP over IPX

The NetBlazer supports SNMP over IPX for use in Novell environments based on Novell recommendations (RFC 1298).

Note: *Telebit recommends the use of the Configuration GUI utility to configure NetBlazers in an IPX network environment.*

snmp authentication on|off

The **snmp authentication** command enables/disables the transmission of an authentication trap by the NetBlazer when a query is received that has an invalid community name, or the community has a host list defined and the sending host is not on the list.

Trap messages are issued only to the members of the trap community.

Note: *Authentication traps are issued only to hosts in a special community (the trap community) designated to receive these traps. This community is established through the community command. If no community exists, traps are not sent.*

snmp community

The **snmp community** commands are used to create and maintain a list of communities in which various management stations or SNMP application entities reside. Permissions of read only, or read/write are assigned on a per community basis. Note that authentication and startup traps are not issued to these communities.

```
NB:Top>Configure>SNMP>Community> ?
Available commands:
add             delete          host            list
top>            up>             ?
```

snmp community add <community> [<ro>|<rw>]

snmp add is used to create a new community in the list of communities. Two privilege levels may be assigned to the added community: <ro> (read only), and <rw> (read/write).

Community names can be up to 31 characters long. Note, however, that if spaces or special characters are used, it is necessary to surround them with double quotes in order for the NetBlazer to see the string as one argument.

The special trap community is created by **not** specifying any [<ro>|<rw>] privilege. Note that only one trap community can exist. A newly added trap community name replaces the previous name.

snmp community delete <community>

snmp delete is the converse of **snmp add**. It deletes a specified community from the list of communities, and all the hosts within that community.

snmp community host <community-name> add|delete <net-type><hostid>

The community host command is used to add or delete a host to the specified community name. The <net-type> is the type of network on which the host resides, IP or IPX. The <hostid> is either the IP or IPX address, or the domain name of the specific host.

snmp community list [<community-name>]

The **community list** command, when used by itself provides a list of existing communities and their permissions (read only, or read/write), and how many hosts are in each. Specifying a community name provides the list of hosts within that community.

```
NB:Top>Configure>SNMP>Community> list
Community                Access       hosts
public                   read-only    0
private                  read-write   0
SNMP_trap                NULL         0
```

snmp enterprise [disable|default|enable [ro|rw]]

The **snmp enterprise** command enables or disables SNMP access to the enterprise MIB. The default condition of the enterprise MIB is enabled with read/write permission enabled, however, the default condition will not be recorded in a save operation.

When disabled, the enterprise MIB appears not to exist. SNMP requests for variables contained in the MIB will receive a return message that the variable does not exist.

Note:

1. *If you plan to use the Telebit Configuration GUI Utility, it is necessary that the enterprise MIB be* enabled *with read\write permission (factory default).*
2. *When the NetBlazer is initially powered on, the default community setting permits* any host *to access the Enterprise MIB. In other words, if no hosts are listed, all hosts are accepted. Be sure to set up your community lists and hosts as part of the initial configuration.*
3. *When initially powered on, the NetBlazer defaults are:*
   - enterprise MIB enabled *with read\write permission*
   - community setting permits any host *to access the Enterprise MIB*

snmp list

The **snmp list** command shows the status and configuration settings for the NetBlazer. More specifically, it lists some of the MIB-II variables that are specific to SNMP. These items are illustrated in the example shown below:

```
NB:Top>Configure>SNMP> list
( 1)snmpInPkts                          0     ( 2)snmpOutPkts                          0
( 3)snmpInBadVersions                   0     ( 4)snmpInBadCommunityNames              0
( 5)snmpInBadCommunityUses              0     ( 6)snmpInASNParseErrs                   0
( 7)                                    0     ( 8)snmpInTooBigs                        0
( 9)snmpInNoSuchNames                   0     (10)snmpInBadValues                      0
(11)snmpInReadOnlys                     0     (12)snmpInGenErrs                        0
(13)snmpInTotalReqVars                  0     (14)snmpInTotalSetVars                   0
(15)snmpInGetRequests                   0     (16)snmpInGetNexts                       0
(17)snmpInSetRequests                   0     (18)snmpInGetResponses                   0
(19)snmpInTraps                         0     (20)snmpOutTooBigs                       0
(21)snmpOutNoSuchNames                  0     (22)snmpOutBadValues                     0
(23)                                    0     (24)snmpOutGenErrs                       0
(25)snmpOutGetRequests                  0     (26)snmpOutGetNexts                      0
(27)snmpOutSetRequests                  0     (28)snmpOutGetResponses                  0
(29)snmpOutTraps                        0     (30)snmpEnableAuthenTraps          disabled
SNMP UDP server is enabled
SNMP IPX server is disabled
sysDescr: Telebit NetBlazer Version 2.0
Telebit enterprise MIB is disabled
Startup traps are disabled
Authentication traps are disabled
```

Note:

1. *Variables 7 and 23 are not used and are therefore blank.*
2. *sysLocation and sysContact do not appear in the list if they are not set.*

snmp startup on|off

The **snmp startup** command activates or deactivates the NetBlazer's issuing of two types of startup traps:

- NetBlazer cold boot.
- NetBlazer warm boot.

Note: *Startup traps are issued only to hosts in a special community (the trap community) designated to receive these traps. This community is established through the community command. No traps are sent if the trap community does not exist.*

snmp sysContact <sysContact-string>

The **snmp sysContact** command sets the string that is transmitted to a remote management station when the sysContact variable is queried. This should be set to the name and phone number of the person responsible for the operation of the NetBlazer to which the query is directed. The default value for <sysContact-string> is an empty string.

snmp sysLocation [<location>]

The **snmp sysLocation** command sets the string that is transmitted to a remote management station when the sysLocation variable is queried. This should be set to the physical and logical location of the NetBlazer to which the query is directed. The default value for <sysLocation-string> is an empty string.

snmp sysName [<name>]

The **snmp sysName** command sets the string that is transmitted to a remote management station when the sysName variable is queried. This should be set to the domain name and/or IP address of the NetBlazer to which the query is directed. The default value for <sysName-string> is an empty string.

By convention (in SNMP), sysName is the full domain name of the device, however you are at liberty to use whatever string is appropriate to your use.

Syslog Commands

The **syslog** command takes you to the Configure Syslog command level. To view a list of the commands available at this level, type a question mark (?) at the Configure Syslog command level.

```
NB:Top> syslog
NB:Top>Configure>Syslog> ?
Available commands:
buffer          console         host            interval
list            session         top>            up>
?
```

These commands determine where NetBlazer syslog messages are sent. The NetBlazer is shipped from the factory with syslog turned off (i.e. no syslog messages are reported). You can configure the NetBlazer to send syslog messages to a UNIX host, the NetBlazer console, your login terminal, and/or an internal buffer in the NetBlazer's memory. Messages stored in memory can be listed to a disk file or to your terminal screen.

Each **syslog** destination command except **syslog host** allows you to filter messages sent to the destination based on the severity level of the messages. Filtering of syslog messages sent to a UNIX host is controlled by the *syslog.conf* configuration file on the UNIX system.

Recognized message severity levels are listed in Table SYS-1 in descending order of severity.

Table SYS-1. Syslog Message Severity Level

Level	Message Severity	
0	emerg	panic conditions requiring immediate attention
1	alert	conditions that should be corrected immediately
2	crit	critical error conditions
3	err	other error conditions
4	warning	warning messages
5	notice	non-error conditions requiring special handling
6	info	informational messages
7	debug	messages that are used only for debugging

The <level> and <=level> parameters in the **syslog** destination commands determine which syslog messages are sent to the specified destination.

If you specify a level without the equal sign (=), all messages at that level and above are sent to the syslog destination. For example, a **syslog console on 5** command directs all syslog messages with a severity level of 0 through 5 to the console.

If you specify a level with the equal sign (=), only messages at that level are sent to the syslog message destination (e.g., **syslog console on =5** sends only syslog message with a severity level equal to 5 to the console).

If you do not specify a severity level, the NetBlazer uses the default value of 7, which sends all syslog messages to the syslog destination (e.g., **syslog console on** directs all syslog messages to the console).

syslog buffer off | on [<level>|<=level>] | list [<file name>] | size <sizeK>

The **syslog buffer** command is used to start or stop sending syslog messages to a buffer in NetBlazer's internal memory, set the size of the buffer, and list the current contents of the buffer to your screen or to a file on diskette.

The default size of the syslog buffer is 20 Kbytes. To increase or decrease the amount of memory allocated to the syslog buffer, issue a **syslog buffer size** <sizeK> command where <sizeK> is the number of Kbytes to use for the syslog buffer.

```
NB:Top>Configure>Syslog> buffer size 40
```

The **syslog buffer on** [<level>|<=level>] command starts sending syslog messages to an internal memory buffer. The optional <level> and <=level> parameters specify the severity level(s) of messages that are sent to the console. Refer to Table SYS-1 for a description of the syslog message severity levels.

You can list the contents of the buffer to your terminal screen or to a file on diskette.

```
NB:Top>Configure>Syslog> buffer list
Tue Mar 16 15:28:06 1993 - root logged-in from 143.191.10.100:1590
Tue Mar 16 15:28:25 1993 - root on 143.191.10.100:1590 at Mar 16 15:28 for
19 seconds
NB:Top>Configure>Syslog> buffer list message.log
NB:Top>Configure>Syslog> more message.log
Tue Mar 16 15:28:06 1993 - root logged-in from 143.191.10.100:1590
Tue Mar 16 15:28:25 1993 - root on 143.191.10.100:1590 at Mar 16 15:28 for
19 seconds
```

Caution: *If you list the buffer to an existing file name, the current contents of that file are overwritten. Choose a file name that is not used by the NetBlazer.*

To stop sending syslog messages to the buffer, enter a **syslog buffer off** command.

Note: *Turning off syslog to the buffer deletes the current contents of the buffer.*

syslog console off | on [<level>|<=level>]

The **syslog console on** [<level>|<=level>] command controls sending syslog messages to the NetBlazer's console, defined by the **setconsole** command. The optional <level> and <=level> parameters specify the severity level(s) of messages that are sent to the console. Refer to Table SYS-1 for a description of the syslog message severity levels.

To stop sending syslog messages to the console, enter a **syslog console off** command.

Note: *To change the severity level of messages sent to the console, you must turn off syslog to the console.*

syslog host off | on [<loghostname>] [<facility>]

The **syslog host** command controls sending NetBlazer activity information to a UNIX system's *syslogd* daemon, which then forwards this information to the appropriate system log files and/or users.

For information on setting up your UNIX host system, see the following UNIX manual pages: *syslog*, *syslogd*, *syslog.conf*.

The <loghostname> parameter identifies which UNIX system to send the NetBlazer activity information. This parameter may be an IP address or a domain host name.

The <facility> parameter specifies which *syslog* facility to use when sending accounting log messages to a host system. Refer to the *syslog.conf* description in your UNIX manual for a list of recognized *syslog* facilities.

The **syslog host on** <loghostname> <facility> command allows the NetBlazer to start sending syslog messages to a specified host and facility.

```
NB:Top>Configure>Syslog> host on napa local0
```

If you enter a **syslog host on** command without specifying a host or facility, the NetBlazer uses the host name and/or facility settings previously defined by the **syslog host on** [<loghostname>] [<facility>] command.

You can verify that syslog messages from the NetBlazer are reaching the right file on the host system by issuing a **msglog** <message> command. The <message> parameter may be a single word or several words enclosed in quotes. Refer to the *NetBlazer Configuration Guide* for additional instructions on how to verify that syslog is set up correctly.

The **syslog host off** command stops sending syslog messages to a UNIX system.

syslog interval flush | off | on [<interval in minutes>]

The **syslog interval on** [<interval>] command configures the NetBlazer to send syslog messages for logged-in users at specific intervals. For example, entering the following command causes the NetBlazer to log messages that show which users were logged in at the end of each 24 hour day.

```
jade:Top>Configure>Syslog> interval on 3600
```

If you do not specify an interval, the NetBlazer uses a default of 60 minutes (one hour).

The **syslog interval flush** command immediately sends syslog messages for each user that is currently logged in to the NetBlazer.

```
NB:Top>Configure>Syslog> interval flush
NB:Top>Configure>Syslog> buffer list
Wed Mar 17 10:25:14 1993 - interval: root logged-in from line17
```

The **syslog interval off** command stops sending syslog interval messages.

syslog list

The **syslog list** command displays the current status for each of the syslog destinations.

```
NB:Top>Configure>Syslog> list
sending syslog messages to 143.191.10.100, facility = local0
syslog to console is off
syslog to buffer is on; level <= 4
syslog internal buffer size = 20K
syslog to session is on; level <= 6
syslog interval is off
(Levels: 0-emerg,1-alert,2-crit,3-err,4-warning,5-notice,6-info,7-debug)
(Counts: 0-0, 1-0, 2-0, 3-0, 4-0, 5-1, 6-9, 7-1)
Syslog requested - 11, ignored - 11, queued - 0, dropped - 0.
```

The previous example shows that all NetBlazer syslog messages are sent to a UNIX host, messages with a severity level of 0 through 6 are sent to your login session, and messages with a severity level of 0 through 4 are saved in the NetBlazer's internal memory buffer.

syslog session off | on [<level>|<=level>]

The **syslog session on** [<level>|<=level>] command controls sending syslog messages to your login session. The optional <level> and <=level> parameters specify the severity level(s) of messages that are sent to the session. Refer to Table SYS-1 for a description of the syslog message severity levels.

Note: *The syslog to session function is only active for your current login session and is automatically turned off when you log out.*

To stop sending syslog messages to your session, enter a **syslog session off** command.

Note: *To change the severity level of messages sent to your session, you must turn off syslog to the session.*

User Commands

The **user** command takes you to the Configure User command level. To view a list of the commands available at this level, type a question mark (?) at the Configure User command level.

```
NB:Top> user
NB:Top>Configure>User> ?
Available commands:
add             changeable      command         crypto
delete          destgroup       domainlookup    homezone
list            load            local           password
privilege       remote          restrict        timelimit
userdomain      top>            up>             ?
```

Commands at this level maintain a list of authorized NetBlazer users.

Types of User Records

The NetBlazer supports the following types of user records:

- **Login** – Every user who logs in to the NetBlazer, either through a telnet session or by calling in from a remote workstation, must have an assigned user ID and password. User login records may be entered manually through the **user add** command or by loading a copy of a UNIX password file, as described under the **user load** command.
- **Dial SLIP** – Authenticates a SLIP packet mode connection that is initiated from a remote site. This type of user record is created by setting up a dynamic SLIP interface using the **dynamic slipdial** or **ipdial** commands.
- **Dial PPP** – Authenticates a PPP packet mode connection that is initiated from a remote site. This type of user record is created by setting up a dynamic PPP interface using the **dynamic pppdial** or **ipdial** commands.
- **Dial ARA** – Authenticates an ARA connection that is initiated from a remote Macintosh system.
- **Dial ACS** – Authenticates an ACS connection that is initiated from a remote PC system running NetBlazer ACS software. You must have the ACS option installed on the NetBlazer system diskette to set up an ACS user.

user add [<userid>]

The **user add** command adds a new user ID to the list of authorized users.

Before adding login users, consider setting up user destination groups for the users. Refer to the **destination** command description for information on creating destination groups.

To add a new login user, enter the command followed by a single word user ID. For example:

```
NB:Top>Configure>User> add joe
NB:Top>Configure>User> list joe
USERID        PASSWORD        CRYPTO          USE          PRIVS     DEST-GROUP CMD
joe                                           login        DT
(PRIVS: S=status, C=config, D=dial-ok,
        T=telnet/rlogin-ok, A=acs-ok, B=ARA-ok, N=from-network)
```

This command creates a new login user with no system configuration or status privileges. The user's dialout, rlogin, and telnet access is not limited to an assigned destination group. Refer to the **user privilege**, **restrict**, and **destgroup** command descriptions for instructions on how to change a user's privileges and restrictions.

To prevent unauthorized use of this user ID, assign a password to this user, using the **user password** command described later in this section.

If you want a user to be automatically connected to a particular IP host or destination, see the **user command** command description for instructions.

If you enter the **user add** command without a user ID, the NetBlazer begins a dialog, prompting you for the required information. See the *NetBlazer Configuration Guide* for instructions on how to set up the various types of user records.

user changeable [on|off]

The **user changeable** command determines whether or not nonprivileged users are allowed to change their password. Entering this command without an on or off parameter shows the current state of this function.

```
NB:Top>Configure>User> changeable
Normal users cannot change their passwords
```

The default setting is to not allow normal (nonprivileged) users to change their passwords.

Issuing a **user changeable on** command allows all NetBlazer users to change their passwords.

```
NB:Top>Configure>User> changeable on
Normal users can change their passwords
```

Issuing a **user changeable off** command forbids nonprivileged NetBlazer users from changing their passwords.

```
NB:Top>Configure>User> changeable off
Normal users cannot change their passwords
```

user command <userid> [<command> ...]

The **user command** command automatically invokes one or more commands when a user logs in to the NetBlazer. For example, the **telnet** command assigned to carol in the following command is automatically invoked when carol logs in to the NetBlazer.

```
NB:Top>Configure>User> command carol telnet napa
```

To assign more than one command to a user's login, you must separate the commands with a semicolon (;) and enclose the commands in quotes as shown in the next example.

```
NB:Top>Configure>User> command carol "telnet napa; logout"
```

You also can enter a **command** <userid> **source** <filename> command to automatically run a series of commands when the user logs in.

Entering a **user command** <userid> without specifying a command clears the command field for that user.

user crypto <userid> [<crypto-key>]

The **user crypto** command assigns a crypto key to a specified user. Entering this command without a crypto key removes the crypto key for that user.

A crypto key should only be assigned to dynamic interface packet mode user IDs, ARA users, or dial-in users who have the capability to respond to a crypto challenge.

user delete [<userid>]

The **user delete** command removes a user ID from the list of authorized users in the NetBlazer's memory.

Notes:

1. *Deleting a user record that was acquired from a remote passwd file does not remove the user from the passwd file. To do this you must edit the passwd file.*
2. *Deleting a PPP or SLIP user record disables dial-in connections through the dynamic interface with the same name as the user record. See the* ***dynamic*** *command description for more information.*

user destgroup <userid> <destination-group>

The **user destgroup** command assigns a destination group to a specified user ID. The destination group must be previously created by a **destination add** command.

Assigning a user to a destination group allows you to restrict that user's dialout, telnet, and/or rlogin access to only those destination that are defined in the user's destination group. Refer to the **user restrict** command description for instructions on how to restrict a user's dialout, telnet, and/or rlogin privileges.

Another advantage of assigning users to a destination group is that users can use the **session connect** command to select a dialout, telnet, or rlogin destination from a list of assigned destinations.

user domainlookup [on|off]

The **user domainlookup** command enables or disables looking for a remote password entry in DNS. Entering this command without specifying on or off displays the current setting of the command.

user homezone <userid> [<zone>]

The **user homezone** command sets or displays the logical zone in which an ARA user appears when dialing in. If no zone name is specified, the current home zone setting for the user is displayed.

The zone name specified must be one of the zones configured for the default AppleTalk interface (the first Ethernet interface configured for AppleTalk). See the **appletalk interface add** command for more information.

Entering a **user homezone** <userid> "" clears the home zone for the specified user.

user list [<userid>]

The **user list** command displays information about the specified user.

If you enter this command without specifying a user, a listing of all known users is displayed. Note that remote users are not listed unless they are logged in to the NetBlazer.

user load [<host:filename>]

The **user load** command loads and merges the user information from the password file specified by the <host:filename> with the current user records in memory.

Before loading a password file into the NetBlazer, make sure that you understand how the NetBlazer will treat each record in the password file. Refer to the *NetBlazer Configuration Guide* for a detailed description of the NetBlazer passwd file format.

Caution: *When a password file is loaded into the NetBlazer, the user records in the password file overwrite any current user records that have user IDs that are identical to the password file's user IDs.*

Verify that the root user ID has system configuration privileges. The root record should have **SC** *listed in its privileges field. If it does not, assign* ***sysconf*** *and* ***sysstat*** *privileges to root.*

Verify that you can still log into the NetBlazer as root by starting a telnet session to the NetBlazer's loopback address (127.0.0.1). If you cannot log in as root, escape out of the telnet session and change the root password by issuing a ***password root*** *<password> command. Then continue the telnet session and verify that you can log in as root.*

user local <userid>

The **user local** command adds an existing remote user to the list of local users.

```
NB:Top>Configure>User> user local mike
User 'mike' now local
```

This allows you to modify and save the user record locally. Note that the user record is not removed from the remote password file, but the local user record takes precedence over the remote user record.

user password <userid> [<password>]

The **user password** command assigns a password to a specified user ID.

If the optional <password> parameter is not entered, you are prompted for the new password.

```
NB:Top>Configure>User> password carol
Changing password for carol
New Password :
Retype Password :
```

Pressing return at both prompts removes the password assigned to the specified user.

Note: *If you want to allow normal (nonprivileged) users to change their passwords, enter a* ***user changeable on*** *command. This allows all users to change their password using the* ***bin passwd*** *command.*

user privilege <userid> [[-]conf] [[-]stat]

The **user privilege** command grants or removes system configuration and/or status privileges for a specified user. Entering conf or stat after the user ID grants the user configuration or status privileges. If either of these two parameters are preceded by a "-" the privilege is removed for the specified user.

user remote [<host>:<filename>]

The **user remote** command sets or displays the location of a remote passwd file. The NetBlazer looks in the remote passwd file for a user ID that is not found in the local passwd file. If the user ID is found in the remote passwd file, the user is logged in and the user's record is temporary added to the list of users in the NetBlazer's memory. This record is removed from the list when the user logs out.

Remote user's record can be changed only on the remote host. However, you can make a remote user record local by issuing a **user local** command. Local user records take precedence over remote user records.

The following command shows you how to configure the NetBlazer to check the /home/NB/passwd file on the remote system named napa for a user ID that is not found in the local user list:

```
NB:Top>Configure>User> remote napa:/home/NB/passwd
remote password file: napa:/home/NB/passwd
```

Because the NetBlazer uses TFTP to read the remote passwd, it must be readable by everyone.

Very large sites can speed up the search for a remote user's record by splitting the remote passwd file into several files based on the first one or more characters in the user ID fields, and specifying a %s in the pathname parameter of the **user remote** command. For example, the following command instructs the NetBlazer to look in a file that has a name that matches the first character of a user's ID.

```
NB:Top>Configure>User> remote napa:/home/NB/passwd/%.1s
remote password file: napa:/home/NB/passwd/%.1s
```

When a remote user named mike attempts to log in, the NetBlazer looks in the /home/NB/passwd/m file on the host named napa for mike's user record.

You can put each user record in a separate file and instruct the NetBlazer to search for a remote user's passwd file by entering a command similar to the following:

```
NB:Top>Configure>User> remote napa:/home/NB/passwd/%s
remote password file: napa://home/NB/passwd/%s
```

In this case, the NetBlazer gets mike's passwd record from a file named /home/NB/passwd/mike on the host named napa.

user restrict <userid> [[-]dial] [[-]telnet]

The **user restrict** command controls whether or not a specified user is restricted to the destinations contained in the user's assigned destination group.

Entering dial or telnet after the user ID restricts the user's dialout or telnet/rlogin privileges to the destinations defined in the user's destination group. Preceding either of these parameters with a "-" removes the specified restriction.

If the user is not assigned a destination group, then the restrictions apply to any dialout, telnet, or rlogin session.

user timelimit <userid> [<minutes>|unlimited]

The **user timelimit** command sets the maximum duration of an ARA session time before automatically disconnecting a remote user. The time limit can be set from 1 to 1440 minutes (one day). The default **timelimit** value for ARA users who are not assigned a specific time limit is unlimited.

A warning is set to the ARA users, who have a timelimit, approximately two minutes before their time has expired.

user userdomain [<domainname>]

The **user userdomain** command tells the NetBlazer which domain suffix to use when looking for a remote passwd entry in DNS. Entering this command without specifying a domain suffix displays the current setting of the **userdomain** command.

TELEBIT®

First In Dial-Up Networking